Name: Konstantinos Giatras
Course: Molecular Modeling of Biomolecules
Midterm Project
Date: 13/12/2022

1. What is the biology of PTEN and why is it important to study it?

PTEN (Phosphatase and Tensin Homolog deleted on Chromosome 10) is a protein that plays a critical role in maintaining cell homeostasis and acts as a tumor suppressor in humans. PTEN's gene is located on chromosome 10q23 and it is mutated in many different types of human cancers and in hereditary cancer predisposition syndromes, with its genetic alterations being among the most frequently noted somatic mutations in a variety of cancers. PTEN also has a role in regulating metabolism and how this regulation leads to different biological outcomes. PTEN's enzymatic activity is primarily responsible for removing phosphate groups from key intracellular phosphoinositide signaling molecules. This activity serves to maintain cell homeostasis and prevent the uncontrolled growth and proliferation of cells, which can lead to the development of cancer. Studying PTEN is important because understanding its function and how it is regulated may lead to the development of new cancer therapies. PTEN's role in regulating metabolism is also an important area of study, as this understanding may lead to new treatments for diseases such as diabetes. It is a vital protein in the regulation of cell growth and proliferation and has a wide range of potential applications in the medical field.

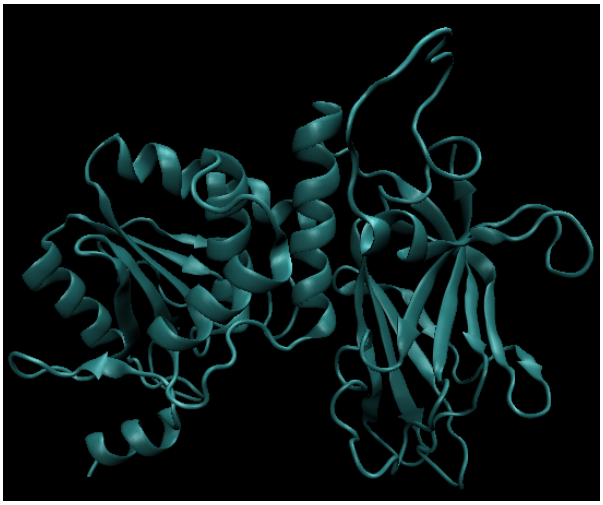
PTEN has both protein phosphatase and phosphoinositide phosphatase activity *in vitro* and is able to remove phosphate groups from intracellular phosphoinositide signaling molecules. This activity normally serves to restrict growth and survival signals by limiting the activity of the PI3K pathway. The amino-terminal region of PTEN is similar to tensin and auxilin, while the carboxy-terminal region is less well understood. It is thought that the C2 domain of PTEN may play a role in positioning the catalytic domain on the membrane and that PTEN's tumor suppressor function may involve the membrane-bound $PI(3,4,5)P_3$ substrate. PTEN's growth suppression activity can be reduced or eliminated by mutations in either of its two domains. Mutation of the phosphatase signature motif can eliminate PTEN's lipid phosphatase activity but not its protein phosphatase activity.

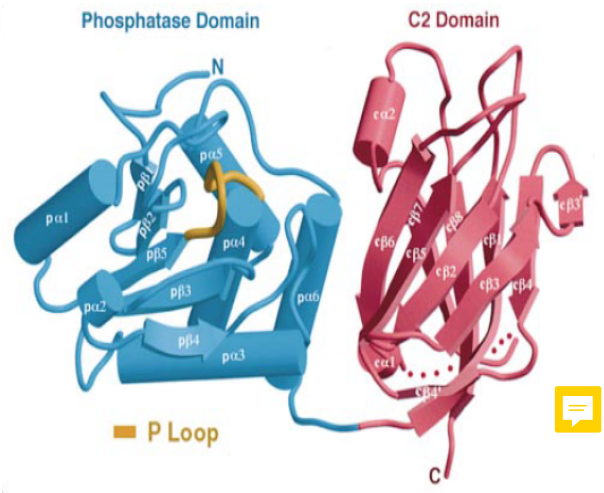
PTEN mutations are commonly found in glioblastoma, with about 50% of cases having a mutated form of the protein. PTEN has potential as a therapeutic target in tissue regeneration and stroke/nerve injuries due to its role as an antagonist of AKT-mediated cell growth and proliferation. It also has potential in Alzheimer's disease, as inhibiting or silencing PTEN leads to reduced cognitive deficits. PTEN is a tumor suppressor protein, so activating it may also be desirable. However, PTEN has proven difficult to target due to the high conservation of its active site among other tyrosine phosphatases.

2. Describe the main structural features of the PTEN protein. Proceed with protein preparation, psf generation, and minimization in NAMD. Explain the procedure you used for setup and provide the energy minimization plot vs minimization steps.

We download the Crystal Structure of the PTEN Tumor Suppressor (1D5R) from the Protein Data Bank as a pdb file. We then load 1d5r.pdb into VMD in order to observe the structural features of the protein. We can identify all three of the common secondary structures in proteins, namely alpha helices, beta sheets, and loops.

According to the literature, PTEN is a protein that contains both a phosphatase domain and a C2 domain. The phosphatase domain is similar to protein phosphatases but has an enlarged active site that is important for the accommodation of the phosphoinositide substrate. The C2 domain of PTEN is able to bind to phospholipid membranes *in vitro* and mutations of basic residues that could mediate this binding can reduce PTEN's membrane affinity and its ability to suppress the growth of glioblastoma tumor cells. The phosphatase and C2 domains of PTEN associate across an extensive interface, suggesting that the C2 domain may serve to productively position the catalytic domain on the membrane. PTEN also has unstructured or loosely folded regions of 7 and 49 residues at the N-terminus and C-terminus, respectively, and a region of 24 residues in an internal loop. Additionally, PTEN contains nine basic and two hydrophobic side chains emanating from the CBR3 and Cα2 elements of the C2 domain, which are on the same face as and in close proximity to the phosphatase active site. These features may be important for PTEN's function and regulation.





<u>Protein Preparation:</u>

To proceed, we need a topology file (top.chm), a parameter file (par.chm), a protein structure file (psf) and a coordinate file (pdb). The CHARMM force field (description of all bonded and non-bonded interatomic forces) provides us with the first two files (in this case, we used a pre-made library of parameter files), while we get the corresponding pdb file for the PTEN protein (1d5r) from the Protein Data Bank, as mentioned before.

We check if there are any missing amino acids and fill them in (using the Maestro software).

We then examine the crystal waters using VMD, and decide whether or not we are going to remove them. To make that choice, we need to refer to the literature, to figure out how important these water molecules are for the structure. In this case, we decided to remove all the crystal waters (we delete the atoms from the pdb file using a text editor).

Psf Generation:

We open VMD, load in the pdb file and we use the Automatic PSF Builder built-in modeling extension that VMD provides to generate the psf file (by adding the rtf topology file). This step also adds the missing hydrogen atoms to the protein crystal structure.

We proceed with solvating the protein, using VMD's TkConsole. The -t option creates the water box dimensions such that there is a layer of water 5 Å in each direction from the atom with the largest coordinate in that direction. We make the box large enough so that the protein does not interact with its image in the next cell if periodic boundary conditions are used.

Next, through the TkConsole, we neutralize the system and add an additional 150mM NaCl to represent a more typical biological environment.

After that, through the TkConsole, we calculate the dimensions (minimum and maximum values of x, y and z) and the coordinates of the center of the water box, so that we can use them in the rest of the scripts (configuration files):

```
{8.402000427246094 43.78300094604492 -2.700000047683716} {61.15800094604492 124.91799926757813 73.86199951171875}
34.709693908691406 84.41574096679688 35.71088409423828
```

Minimization:

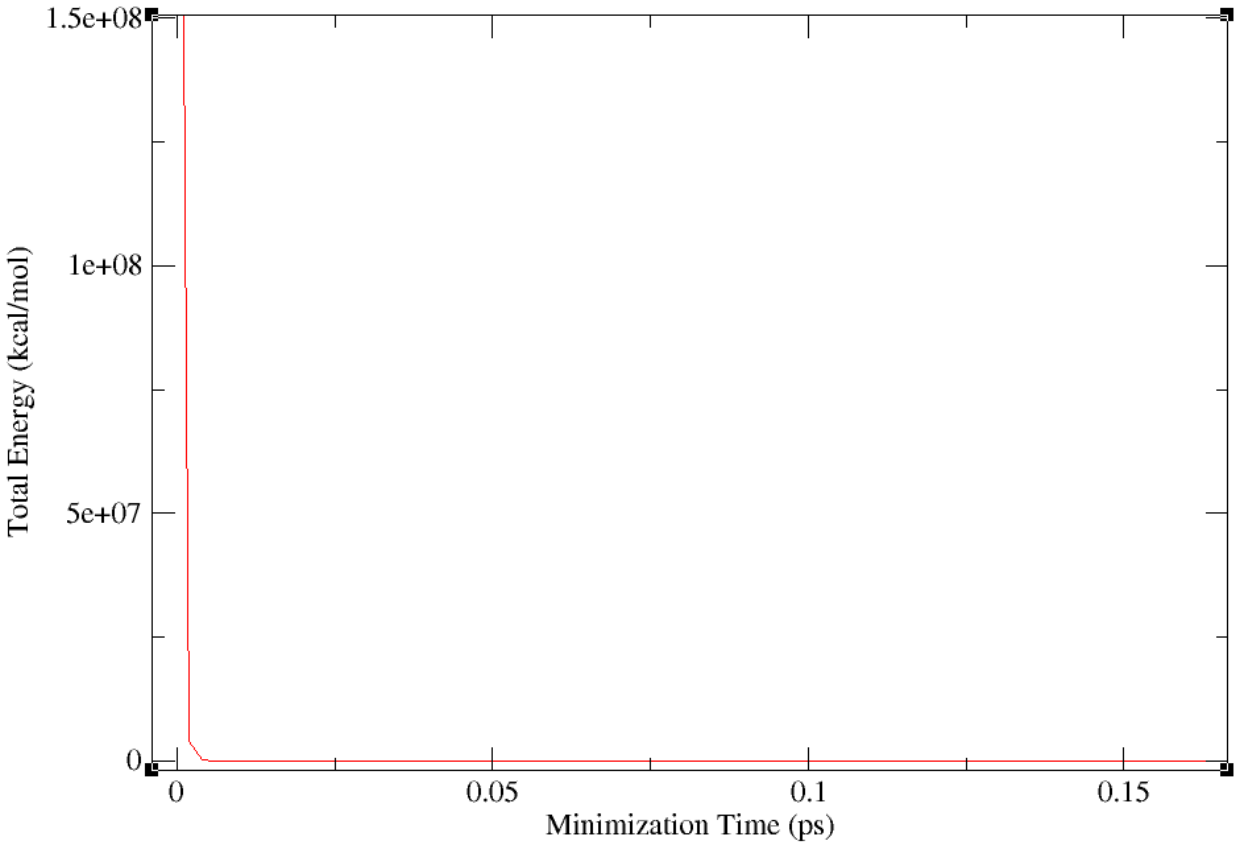
Energy minimization involves searching the energy landscape of the molecule for a local minimum, i.e., the place in which the molecule is relaxed, by systematically varying the positions of atoms and calculating the energy. Energy minimization should remove any unwanted high-energy interactions that may have been present in the crystal structure.

We start by configuring our input (conf) file, so that it has all of the correct parameters (input/output file names, correct parameter files, water box dimensions and center coordinates etc.), in order to run the minimization. We use periodic boundary conditions (surrounding the system under study with identical virtual unit cells. The atoms in the surrounding virtual systems interact with atoms in the real system. These modeling conditions are effective in eliminating surface interaction of the water molecules and creating a more faithful representation of the *in vivo* environment than a water sphere surrounded by vacuum provides) and, in this case, 5000 minimization steps.

We run the minimization from the terminal, using namd2, the conf file as input, and get a log file as output. From the terminal, we grep the 'ENERGY:' from the log file into a dat file (1d5r_wb_mini.dat), which we edit appropriately so that it has defined columns.

Finally we use XMGrace to plot the Total Energy vs Minimization Time. We import the dat file as ASCII block data, and set the 2nd column (minimization time steps, which we convert to ps) as X and the 12th column (Total Energy) as Y.

Total Energy vs Minimization Time:



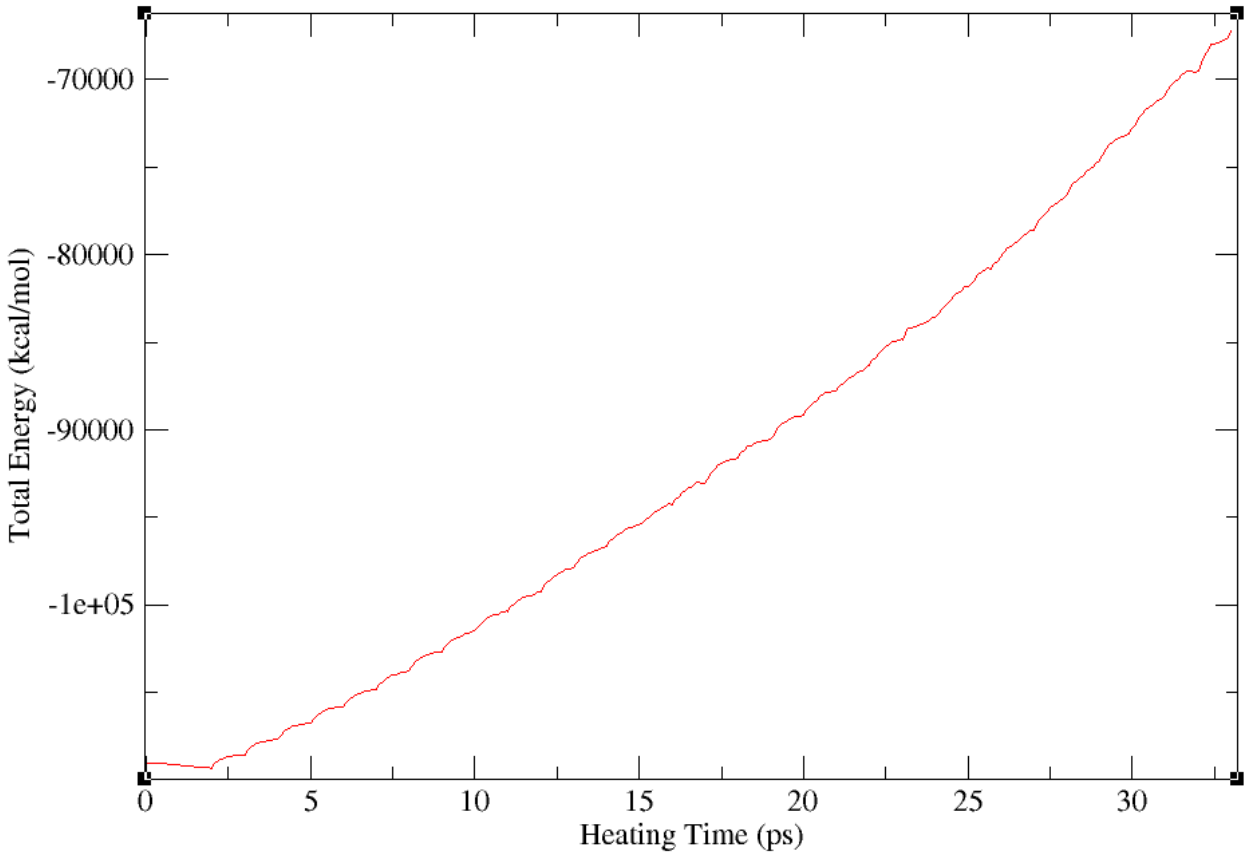
From the plot, we can see that the minimization has converged.

3. Proceed with heating and equilibration of the structure as in the tutorial, always restarting from your restart files produced by NAMD in the previous step. Plot the energy vs time.
4. Assess whether the protein has reached equilibrium. If not, restart the equilibration until you have reached a plateau in the potential energy of the system.

Heating:
To simulate the system's motion over time, the initial velocities of each atom in the system at low temperature are assigned and integrated using Newton's equations of motion. The simulation is begun by assigning new velocities at slightly higher temperatures in a periodic manner during the heating phase, and allowing the simulation to continue until the target temperature is reached. This process is repeated with the initial velocities being set at progressively higher temperatures.

After running the heating (for 500 simulation steps every 10 K, starting from 0 K and ending at 310 K) and generating the 1d5r_wb_heat.dat file, we use XMGrace to plot the data. We use the 2nd column (heating time steps) on the x-axis and 12th column (total energy) on the y-axis of the plot.

Total Energy vs Heating Time:

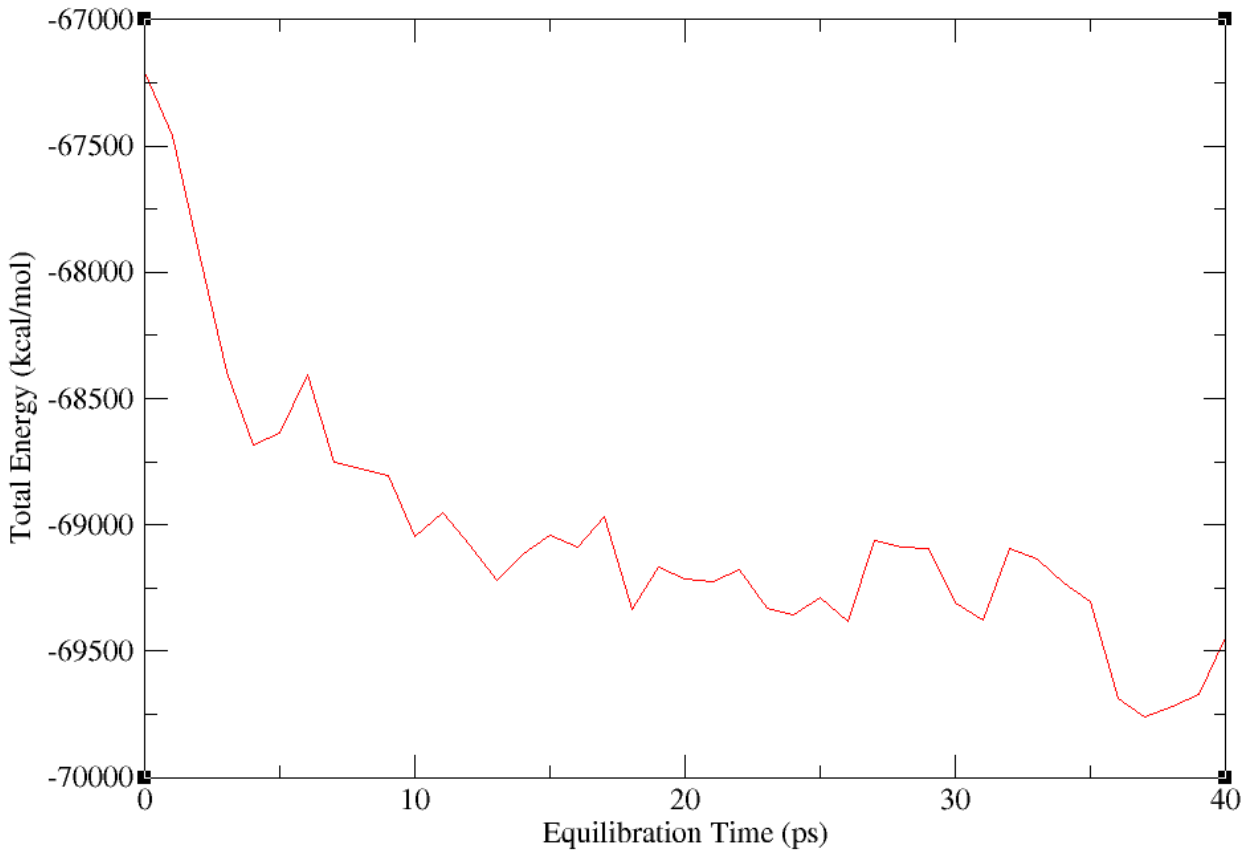


Equilibration:

The equilibration phase of the simulation involves solving Newton's Second Law for each atom in the system to determine its trajectory using molecular dynamics. The simulation is continued once the desired temperature is reached, and during this phase the stability of properties such as structure, pressure, temperature, and energy with respect to time are monitored. If the temperature deviates significantly from the desired value during this process, the velocities are scaled to bring the temperature back to the target value.

After running the equilibration (for 500 simulation steps every 2 fs, for 40 ps) and generating the 1d5r_wb_eq.dat file, we use XMGrace to plot the data. We use the 2nd column (equilibration time steps) on the x-axis and 12th column (total energy) on the y-axis of the plot.

Total Energy vs Equilibration Time:



The system appears to have reached thermodynamic equilibrium (the plot has reached a plateau), therefore there is no need to restart the equilibration. We will confirm this after running the production dynamics.

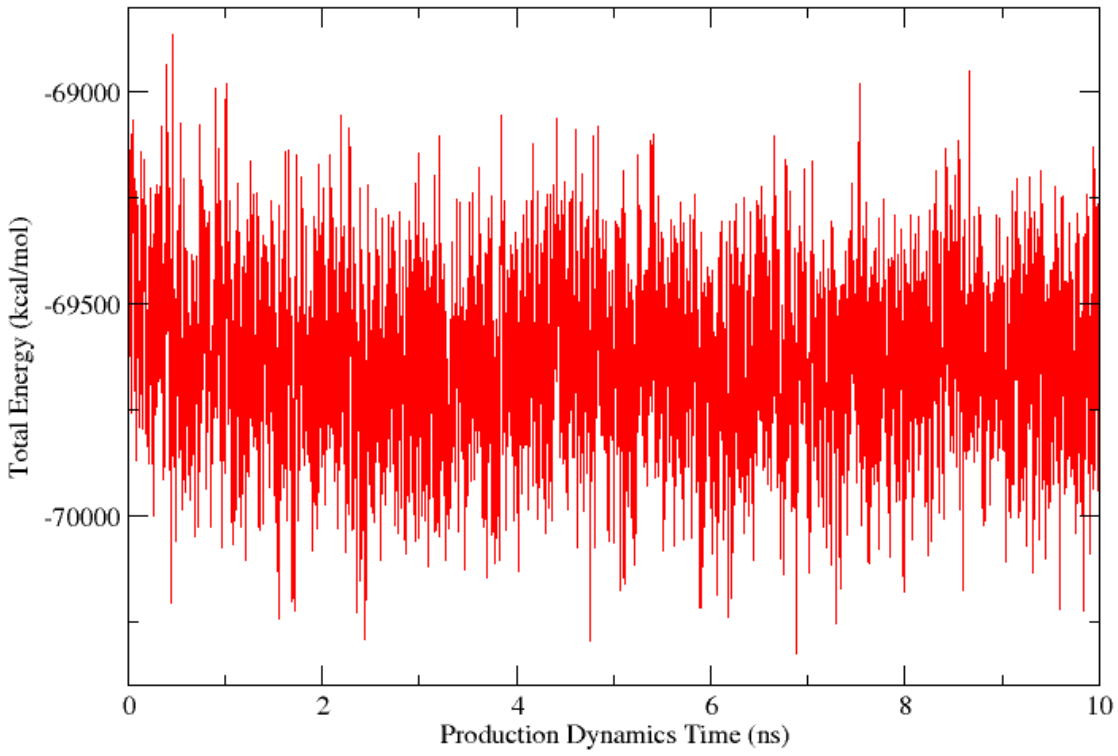
5. Simulate the protein in a production run for at least 10 n

Production Dynamics:

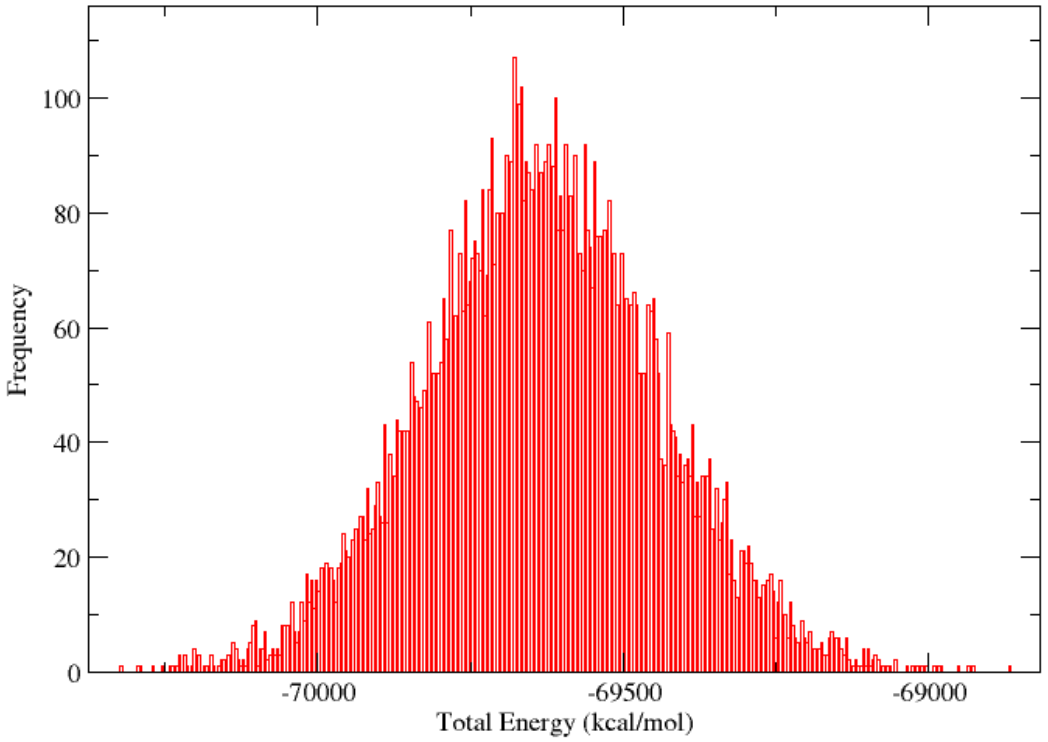
The production phase is the final step of the simulation, during which the system is simulated for a specified time period ranging from hundreds of picoseconds to nanoseconds or longer, with no heating or temperature scaling, to collect data. Coordinates of the system at various points in time are recorded in the form of trajectories (in this case, in the trajectory file 1d5r_wb_prod.dcd) and used to calculate various properties such as mean energy and root mean square fluctuations between structures. Additionally, time-dependent properties such as correlation functions can be derived from these simulations and compared to spectroscopic measurements.

After running the production dynamics (for 500 simulation steps every 2 fs, for 10 ns) and generating the 1d5r_wb_prod.dat file, we use XMGrace to plot the data. We use the 2nd column (production dynamics time steps) on the x-axis and 12th column (total energy) and the 13th column (temperature) on the y-axis of each corresponding plot.

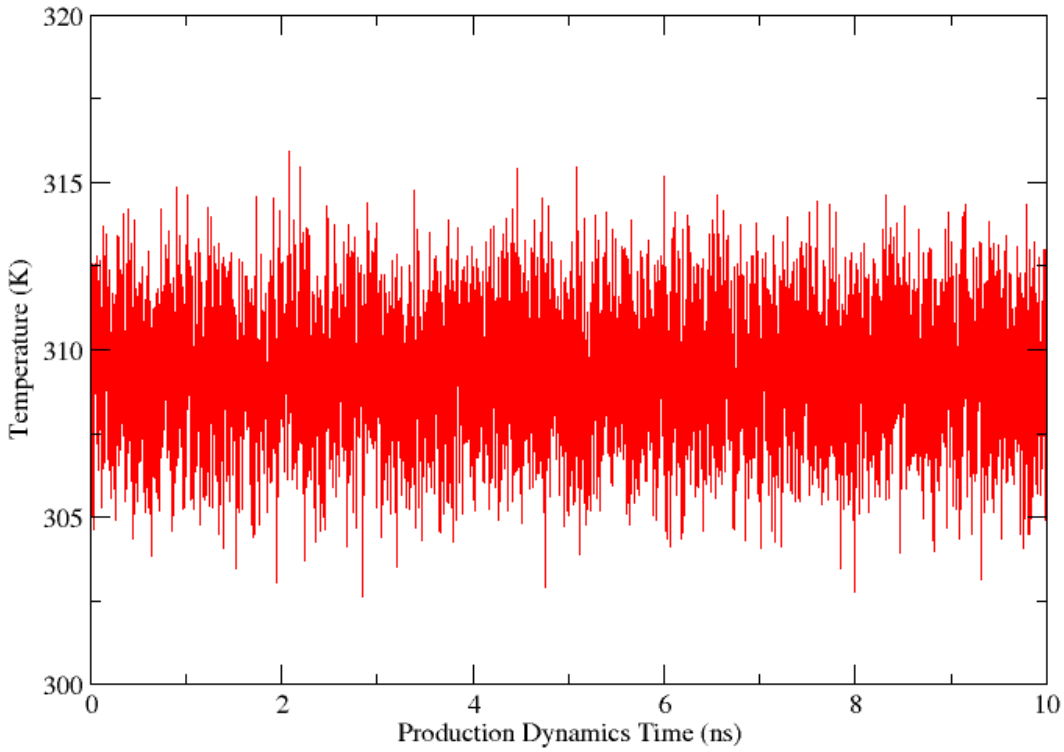
Total Energy vs Production Dynamics Time:



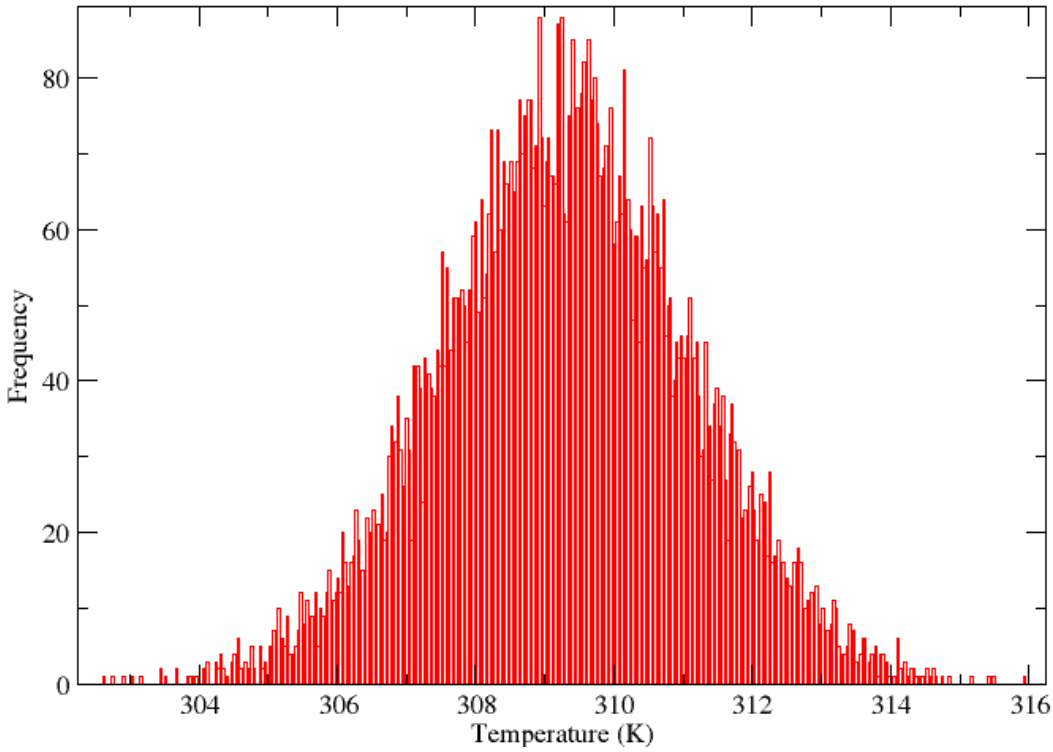
Production Dynamics Total Energy Histogram:



Temperature vs Production Dynamics Time:



Production Dynamics Temperature Histogram:



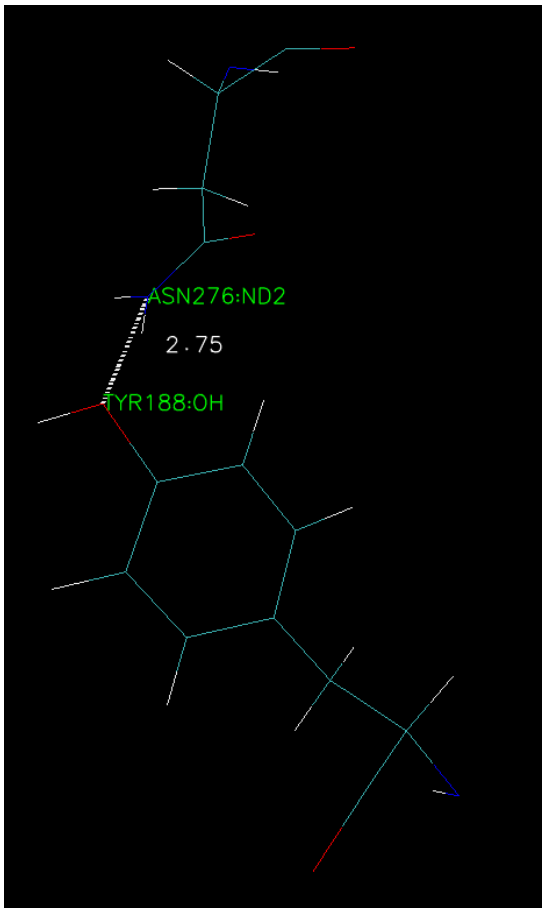
The fluctuations that we observe in the Total Energy vs Production Dynamics Time and Temperature vs Production Dynamics Time plots are within the statistical variance occurring at equilibrium. Therefore, the system remains stable throughout the production dynamics run. This is also corroborated by the histograms, which have the functional form of Gaussian distributions, something that occurs when the system is in equilibrium.

6. Analyze the trajectories.

Trajectory Analysis:
Trajectory analysis of a protein refers to the process of examining the movements and changes in conformation of the protein over time during a production dynamics simulation. This can involve calculating various properties such as the root mean square deviation (RMSD) of the protein's structure, analyzing the mobility of individual residues, and identifying key conformational changes that the protein undergoes. The resulting data can provide insights into the function and behavior of the protein, as well as inform the design of drugs or other molecules that interact with the protein.

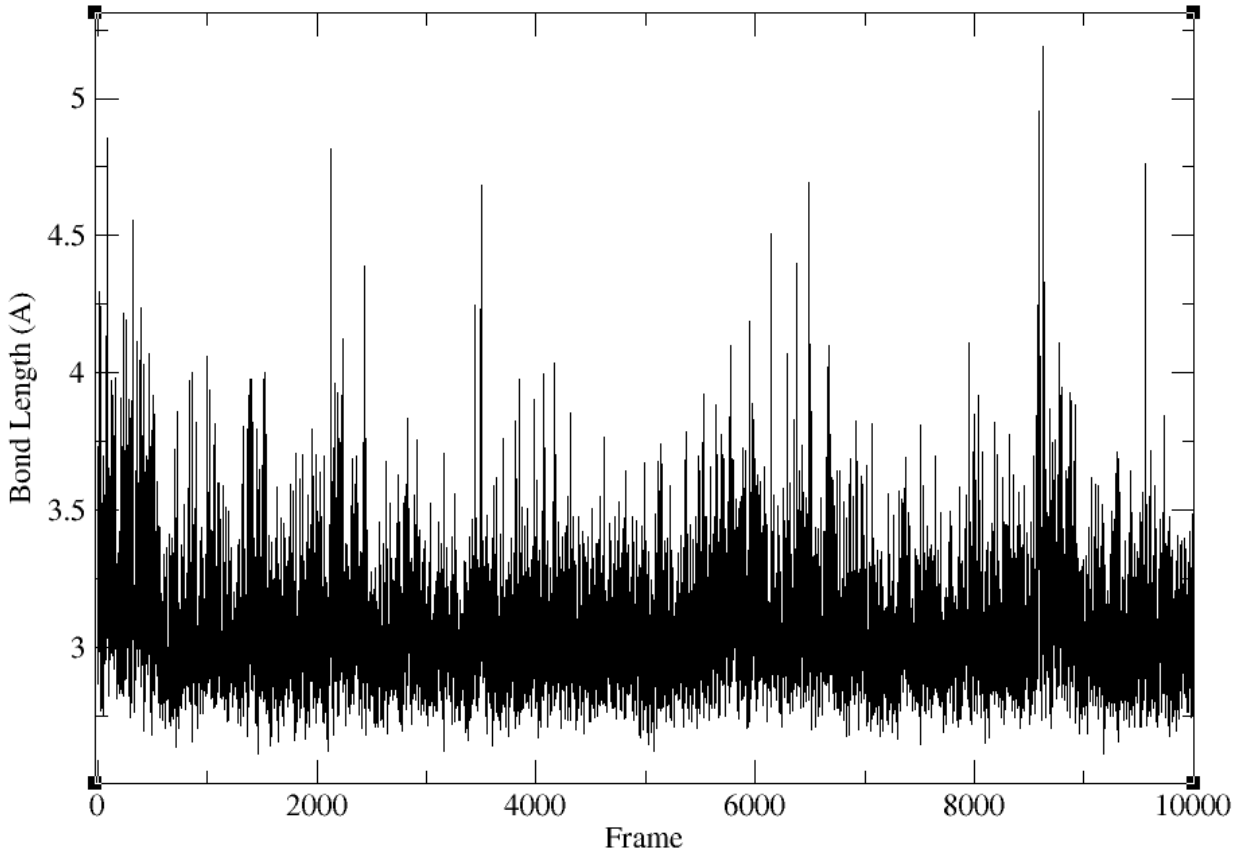
a. Describe which parts (backbone or side chain? Which atoms?) between the amino acids N276 and Y188 form a hydrogen bond between them and plot the time series of this hydrogen bond throughout the trajectory.

We load the ionized.psf file along with the 1d5r_wb_prod.dcd trajectory file (which contains 10,000 frames) into VMD. Inside the selected atoms text box in the graphical representations tab, we type “resid 188 or resid 276” to view the N276 and Y188 amino acids. We can observe that the parts that form a hydrogen bond between them are the nitrogen from asparagine’s side chain with the oxygen from tyrosine’s side chain.



We plot the time series of this hydrogen bond throughout the trajectory using the graph tab inside Graphics → Labels → Bonds and export it into XMGrace:

TYR188:OH/ASN276:ND2 Hydrogen Bond Time Series:



We save the different distance values and calculate the average value and standard deviation using Excel:

Average: 3.068 Å
Standard Deviation: 0.231 Å

The hydrogen bond distance criterion (on average 2.7-3.3 Å) is mostly satisfied throughout the trajectory for the chosen bond.

b. Produce a movie of your simulation and send us the file.

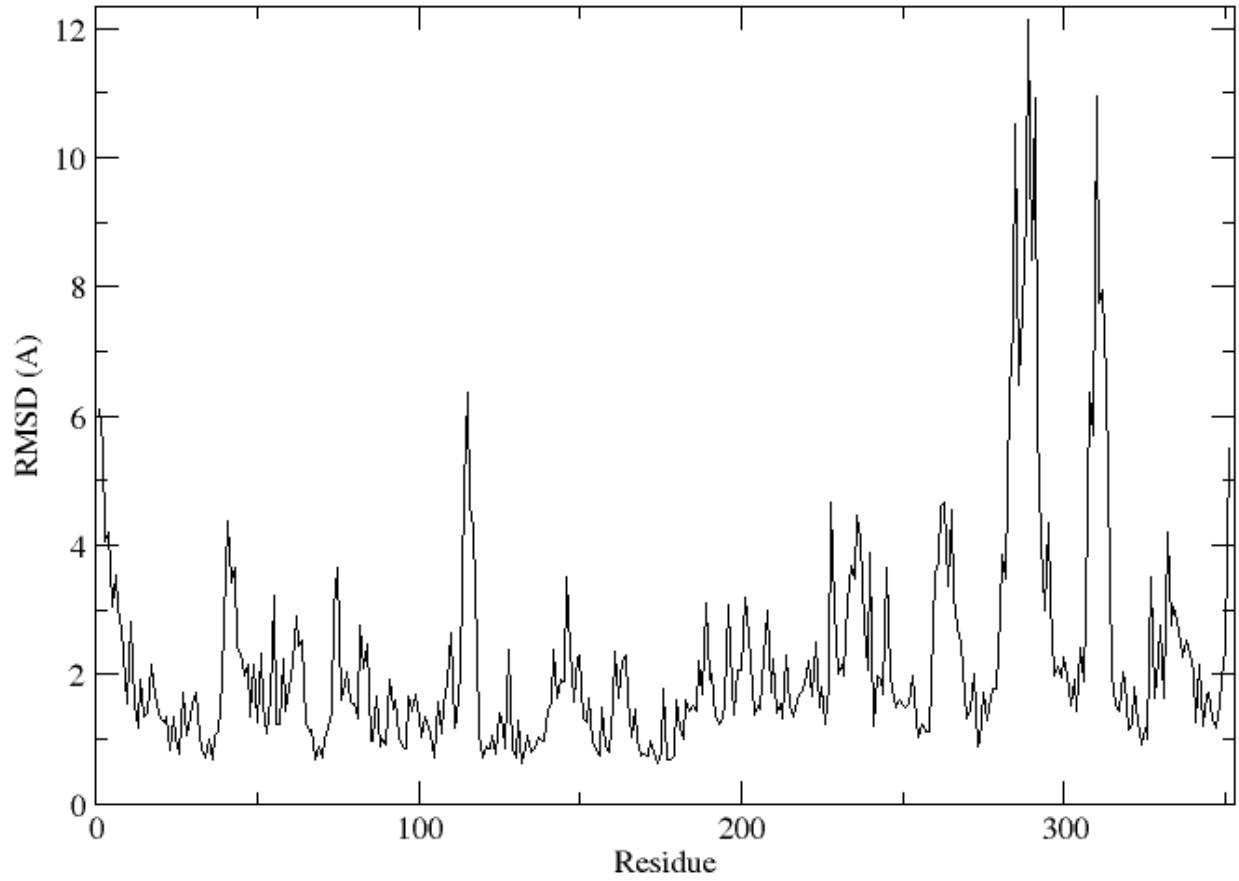
To produce a movie of the production dynamics simulation, we use VMD's Movie Maker, which can be found in the Extensions → Visualization menu. After configuring the colors and display, in order to have the desired visual result, we set up the Movie Maker (Movie Settings: Trajectory, Format: ppmtompeg, Trajectory step size: 7) and render the video, which we later convert to mp4. The movie is attached to the same email as this pdf file.

c. What is the RMSD time series of PTEN? What is the RMSD of the C-ter of the protein?

In computational biology, the Root Mean Square Deviation (RMSD) of atomic positions is the measure of the average distance between the atoms (usually the backbone atoms) of superimposed proteins. It is used as a quantitative measure of similarity between two or more protein structures. Its value is expressed in length units, most commonly Ångström (Å) in structural biology. The lower the RMSD, the more similar (and less mobile) the proteins are.

First, we want to calculate the RMSD of every residue in the protein, so we can get information about the mobility of every part of the protein, by observing the RMSD distribution. The RMSD of each of the 351 residues of PTEN is calculated using a script within VMD, and the resulting values are assigned to the User field for each residue. The script is run by typing “source residue_rmsd.tcl” in the TkConsole window, and then calling the procedure “rmsd_residue_over_time” with the molecule and a list of residue numbers as arguments. The resulting data is printed to a residue_rmsd.dat file and used to color the protein based on the mobility of the residues. We plot the RMSD value per residue using XMGrace:

PTEN Residue RMSD:



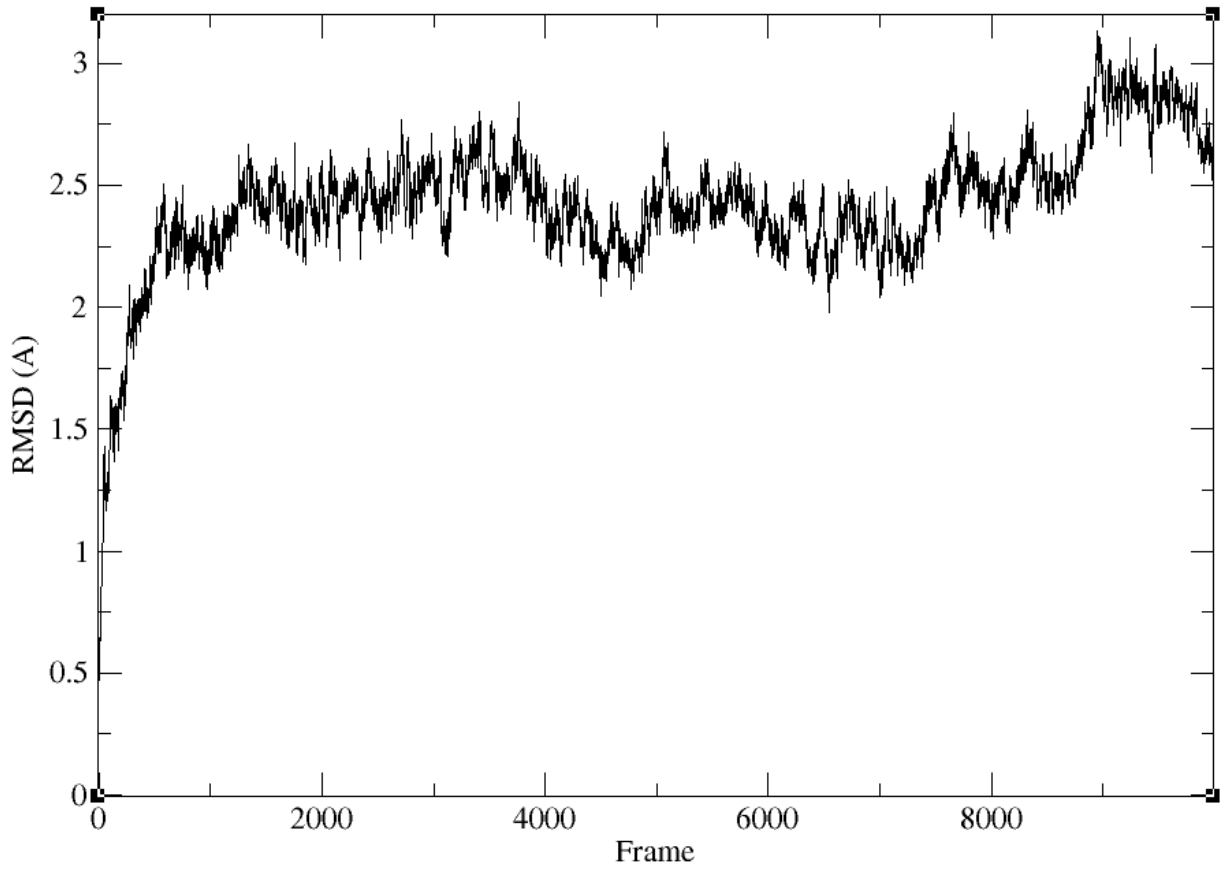
We import the dat file into Excel and calculate the average value and standard deviation:

Average: 2.194 Å
Standard Deviation: 1.688 Å

For RMSD, any value below 2 Å is considered a minimal fluctuation. From our data, we can observe substantial mobility for most of PTEN's residues, especially at the C-terminal domain, which consists of residues 186 to 351 (166 residues).

After that, we want to plot the RMSD time series of PTEN and the C-terminus of PTEN and compare the two. To do that, we use the RMSD Trajectory Tool, another VMD Extension. We select all the alpha carbons (backbone) in the entire protein in the first case, and in residues 186 to 351 in the second case. We align the frames using frame 0 as a reference and plot the RMSD time series and export both plots into XMGrace:

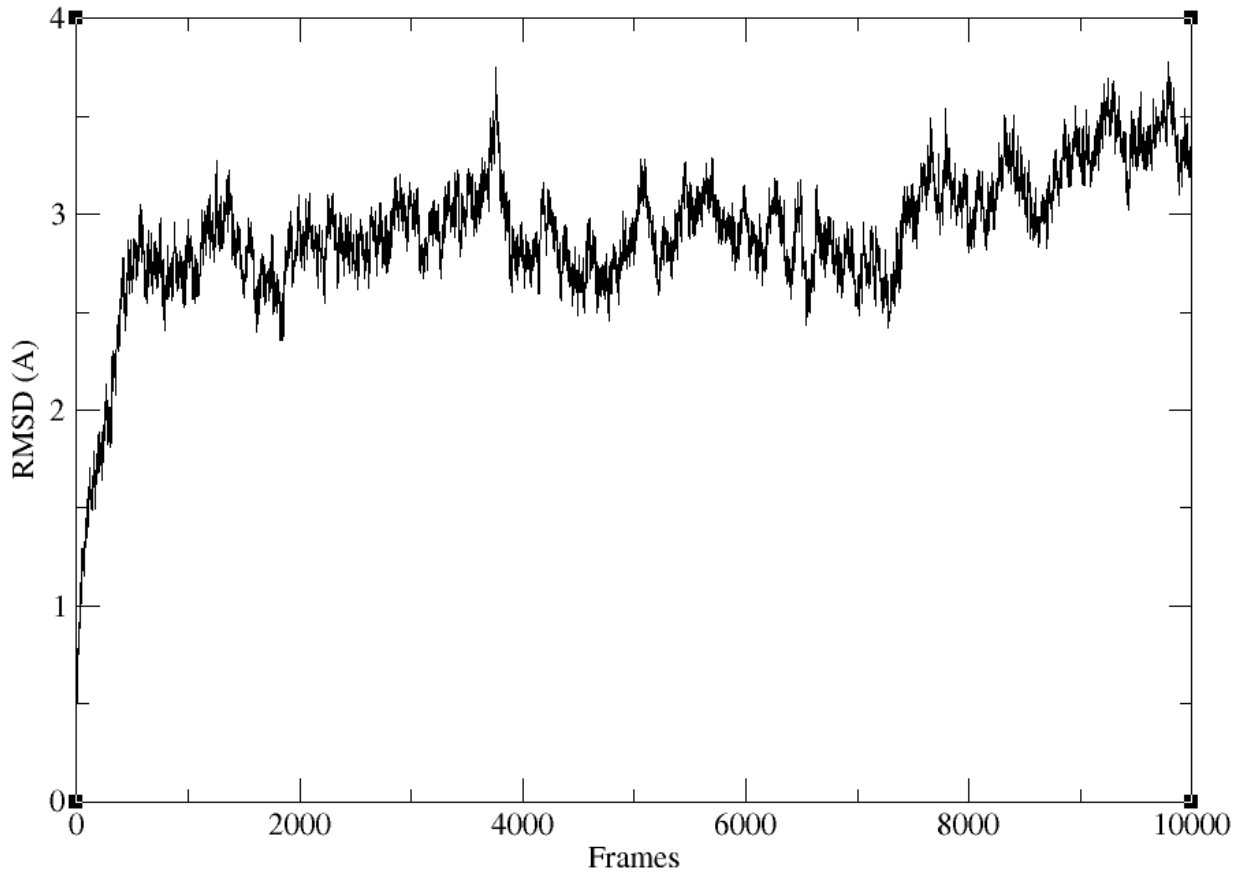
PTEN RMSD Time Series:



The RMSD Trajectory Tool itself gives us the average value and the standard deviation of our data each time:

Average: 2.421 Å
Standard Deviation: 0.262 Å

PTEN C-Terminal Domain RMSD Time Series:



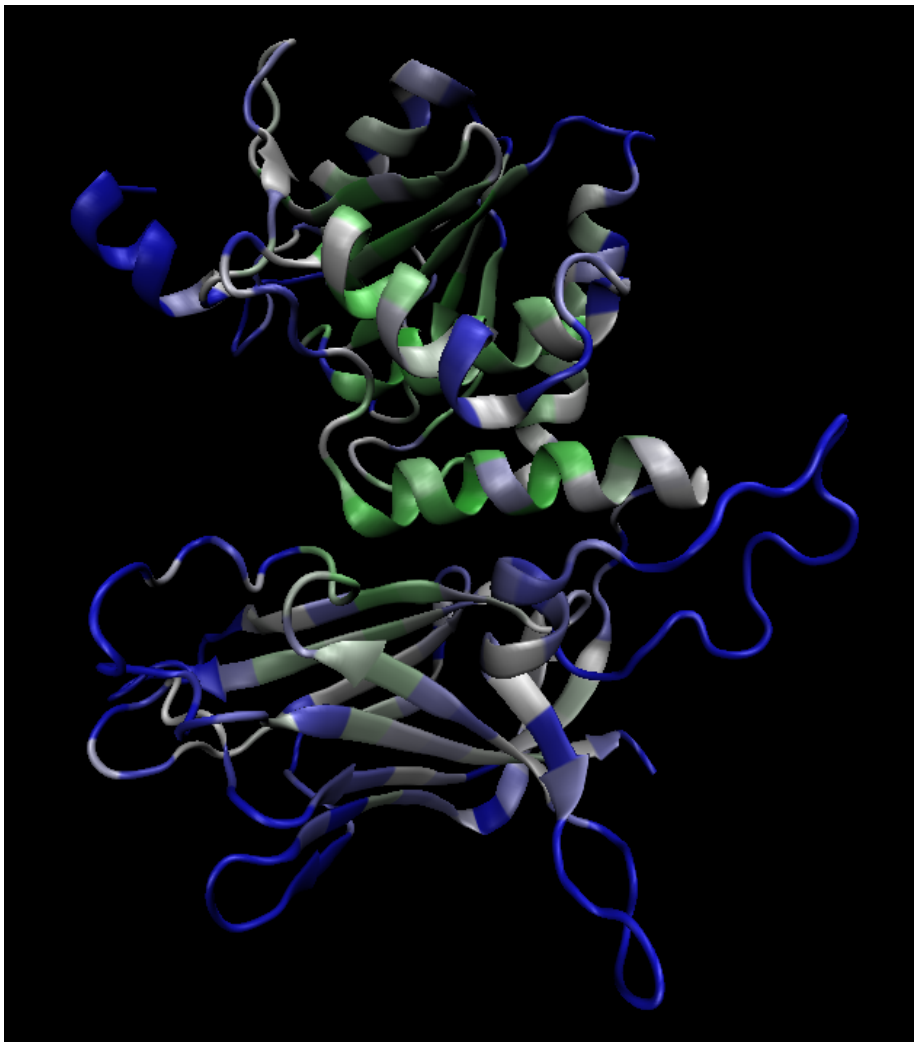
Average: 2.907 Å
Standard Deviation: 0.352 Å

From these results, we can observe that PTEN is quite mobile, especially when it comes to its C-terminus. In addition, the biggest spikes in the PTEN Residue RMSD plot belong to the most flexible parts of the protein. We discuss more about the protein flexibility below.

d. Which part of the molecule is more flexible and why?

The previous procedure also set the value of the User field of all the atoms of the residues in the selection to the computed RMSD value, as mentioned before. We now color the protein according to this value, to recognise which residues are free to move more and which ones move less during the equilibration. In order to use these new values to learn about the mobility of the residues, we create a representation of the protein colored by User value.

We choose NewCartoon as DrawingMethod, and in the Coloring Method drop-down menu, we choose Trajectory > User. In the Color Scale Data Range, we type 0.40 and 2.50 and click Set. We now see the protein colored according to average RMSD values. The residues displayed in blue are more mobile while the ones in green move less.



We observe that the residues on the outside of the protein as well as loops tend to be more flexible, meanwhile the residues on the inside of the protein tend to be less mobile. This can also be seen in the movie that we created.

Proteins are composed of chains of amino acids that fold into specific three-dimensional structures. The outer parts of the proteins, including the loops, are often more flexible because they are not as constrained by the overall structure of the protein. They are also more likely to be exposed to the solvent, which can cause them to be more flexible. Additionally, the loops can sometimes act as hinges, allowing the protein to move or change shape. This flexibility is often important for the function of the protein.

Indeed, our combined results are corroborated by the existing literature. The structure of the phosphatase domain of PTEN (residues 7 to 185) consists of a central beta sheet with two alpha helices on one side and four on the other, which is similar to another dual specificity phosphatase called VHR. However, PTEN has two insertions that are absent in VHR and the tyrosine phosphatase PTP1B, which alter the structure of the active site pocket. This pocket is wider and deeper in PTEN than in the other two proteins, and is able to bind a larger substrate called PI(3,4,5)P_3. The pocket's depth and width are believed to be important for determining the specificity of the protein for certain phospho-amino acids, and the pocket's extension in PTEN is essential for its activity on PI(3,4,5)P_3. The active site of PTEN contains several key residues that are involved in catalysis and substrate binding, and the protein's mobility may be regulated by its interaction with other domains or proteins.

The C2 domain of PTEN has a beta sandwich structure that is similar to the C2 domains of other proteins such as PLCδ1, PKCδ, and phospholipase A2. However, unlike these other C2 domains, the PTEN C2 domain does not bind calcium and is therefore unlikely to bind membranes in a calcium-dependent manner. Despite this, the PTEN C2 domain has several features that are similar to the membrane-interacting regions of calcium-dependent C2 domains, including a CBR3 loop with a net positive charge and solvent-exposed hydrophobic residues, as well as a basic patch on the adjacent alpha helix. These features, along with the fact that the C2 domain has affinity for phospholipid membranes *in vitro*, suggest that the PTEN C2 domain may still have a role in membrane association. The C2 domain interacts with the phosphatase domain of PTEN at an interface that is targeted by tumorigenic mutations, and the C2 domain's mobility may be regulated by its interaction with other domains or proteins.

The motions described above can also be visualized by performing Normal Mode Analysis (NMA) of the first few modes of PTEN. NMA is a computational method used to study the collective motion of atoms in a molecule or system. It involves finding the harmonic oscillations of a system around its equilibrium conformation and describing them using normal modes. Prior to performing NMA, energy minimization is often performed to remove any structural artifacts and ensure that the molecule is in a stable conformation, as well as to make the analysis more computationally tractable. The normal modes of a protein can provide insights into its function by revealing how the protein can move and change conformation, and low-frequency modes are generally more informative because they correspond to more collective motion of the system as a whole. High-frequency modes correspond to more local motion of individual parts of the system and are less informative about protein function. NMA is advantageous over molecular dynamics simulations because it is less computationally intensive and can provide a more detailed picture of the collective motion of a protein.

e. Calculate the 1st cluster representative (most populated structure) from the trajectory.

Clustering is a way to identify groups (or clusters) of structures in the trajectory that are similar to each other. There are several methods that one can use to perform clustering on a protein trajectory. Hierarchical clustering is a method of clustering that creates a hierarchy of clusters, with each cluster being split into smaller clusters until each individual structure is in its own

cluster. This can be a useful method to use when analyzing a protein trajectory, because it allows us to see the relationships between different clusters at different levels of the hierarchy. For the similarity measure, we use RMSD, which measures the deviation of the atoms in each structure from the mean structure of the cluster.

We write a code that finds the 1st cluster representative (cluster with the lowest RMSD and most populated structure within that cluster) from our protein trajectory using hierarchical clustering (see attached).

This code first extracts the protein backbone atoms from the trajectory, then calculates the pairwise RMSDs for all frames in the trajectory using mdtraj. It then performs hierarchical clustering on the pairwise RMSD distances using scipy.cluster.hierarchy, and extracts the cluster assignments for each structure in the trajectory using the fcluster function.

Next, it calculates the average RMSD for each cluster by iterating over the clusters and finding the mean RMSD of all the structures in each cluster. It then finds the cluster with the lowest average RMSD using numpy.argmin, and finds the structure with the lowest RMSD within that cluster using numpy.argmin again. Finally, it accesses the representative structure in the original trajectory using the index of the structure with the lowest RMSD within the cluster.

After running the code, we can see that the structure with the lowest RMSD belongs to frame 37 of the trajectory:

```
# Find the cluster with the lowest RMSD
lowest_rmsd_cluster = np.argmin(cluster_rmsds)

# Check if the cluster is not empty
if len(cluster_assignments[cluster_assignments == lowest_rmsd_cluster]) > 0:
    # Find the structure with the lowest RMSD within the cluster with the lowest RMSD
    lowest_rmsd_structure = np.argmin(distances[cluster_assignments == lowest_rmsd_cluster])
    # Access the structure in the original trajectory
    representative_structure = protein_traj[lowest_rmsd_structure]

    # Print the structure with the lowest RMSD and the corresponding frame of the original trajectory
    print("Structure with lowest RMSD:", lowest_rmsd_structure)
    print("Frame in original trajectory:", representative_structure)
else:
    # If the cluster is empty, print a message and set the representative structure to None
    print("No structures found in cluster with lowest RMSD")
    representative_structure = None
```

```
Structure with lowest RMSD: 37
Frame in original trajectory: <mdtraj.Trajectory with 1 frames, 1404 atoms, 351 residues, and unitcells>
```

References:

- https://www.rcsb.org/structure/1D5R
- https://www.cell.com/cell/fulltext/S0092-8674(00)81663-3?_returnURL=https%3A%2F%2Flinkinghub.elsevier.com%2Fretrieve%2Fpii%2FS0092867400816633%3Fshowall%3Dtrue
- https://www.frontiersin.org/articles/10.3389/fendo.2018.00338/full
- https://www.researchgate.net/profile/Isabelle-Sansal/publication/8455597_The_Biology_and_Clinical_Relevance_of_the_PTEN_Tumor_Suppressor_Pathway/links/588a08e74585157012036763/The-Biology-and-Clinical-Relevance-of-the-PTEN-Tumor-Suppressor-Pathway.pdf
- https://www.sciencedirect.com/science/article/pii/S0092867408005047
- https://www.sciencedirect.com/science/article/abs/pii/S0014482700951309