

DEVELOPMENT OF THE FAST ANALYSIS SUITE AND ITS IMPLEMENTATION ON  
ANALYSIS OF MHC CLASS I PROTEINS AND KRAS

A Thesis by

Micah E. Heikes

Bachelor of Science, Wichita State University, 2020

Submitted to the Department of Chemistry  
and the faculty of the Graduate School of  
Wichita State University  
in partial fulfillment of  
the requirements for the degree of  
Master of Science

May 2024

© Copyright 2024 by Micah E. Heikes

All Rights Reserved

DEVELOPMENT OF THE FAST ANALYSIS SUITE AND ITS IMPLEMENTATION ON  
ANALYSIS OF MHC CLASS I PROTEINS AND KRAS

The following faculty members have examined the final copy of this thesis for form and content and recommended that it be accepted in partial fulfillment of the requirement for the degree of Master of Science with a major in Chemistry.

---

Katie Mitchell-Koch, Committee Chair

---

Haifan Wu, Committee Member

---

Douglas English, Committee Member

---

David Eichhorn, Committee member

---

Gisuck Hwang, Committee Member

Science is not just a body of facts, but a way of thinking critically and objectively about the world around us.

-Marie Curie

## ACKNOWLEDGEMENTS

I would like to thank my family for their support and encouragement, which have played a significant role in my journey. Their faith in me has been a source of strength and inspiration. I also thank Dr. Katie Mitchell-Koch for her guidance and mentorship, which have been helpful in my academic and personal growth. Her wisdom, patience, and expertise have influenced my approach to research and learning. Additionally, I appreciate my lab mates, Ryan and Eli, for their collaboration and friendship. Their insights and support in the lab have contributed to my successes and helped me overcome many challenges. Lastly, I thank the Wichita State University Chemistry Department for providing a good environment, resources, and opportunities that have aided in my development as a scientist. A portion of this material is based upon work supported by the National Science Foundation under Grant Nos. CHE-1665157 and CHE-2140825 and National Institutes of Health grant 1R15GM146382-01.

## ABSTRACT

This thesis introduces the "Fast Analysis Suite," a novel computational tool designed to revolutionize the analysis of protein structures and dynamics. The suite comprises two main applications: a Structure Analysis Suite and a Bash Script Generator for GROMACS simulations. The Structure Analysis Suite provides comprehensive insights into protein structures, detailing secondary structures, amino acid properties, and solvent accessible surface areas, which are essential for understanding protein functionality and interactions. The Bash Script Generator automates the creation of scripts necessary for advanced molecular dynamics analyses, such as characterizing the biomolecular solvation layer by calculating Diffusion Coefficients, Reorientation Times, Hydrogen Bond Lifetimes, and Radial Distribution Functions, thereby significantly reducing manual scripting efforts and potential errors. The effectiveness and efficiency of this suite are demonstrated through simulations on two biologically significant proteins: MHC Class 1 proteins, which play a critical role in the immune response, and KRAS, a key protein often implicated in cancer development due to its mutational propensity. The results showcase the suite's ability to provide fast, accurate, and comprehensive analyses, highlighting its potential as a valuable tool in protein research and its contribution to the fields of computational biology and molecular biophysics. This work not only presents a significant advancement in the methodology of protein analysis but also paves the way for accelerated discoveries related to hydration dynamics and protein function.

## TABLE OF CONTENTS

Chapter	Page
1. Introduction.....	1
1.1 Hydration Shell Background.....	1
1.2 The Need for a Reproducible and Fast Analysis Protocol.....	4
2. Overview of Fast Analysis Suite.....	6
2.1 Structure Analysis Suite.....	6
2.1.1 Solvent Accessible Surface Area.....	7
2.1.2 Secondary Structure Analysis .....	8
2.1.3 Detailed Residue Analysis.....	10
2.2 Analysis Script Generator.....	10
2.2.1 Navigating the Analysis Script Generator.....	12
2.2.2 Solvation Folder.....	13
2.2.3 Diffusion Folder.....	16
2.2.4 MSD Folder.....	21
2.2.5 RDF Folder.....	22
2.2.6 Dipole Folder.....	25
2.2.7 Dipole Plots Folder.....	29
2.2.8 HBond Folder.....	29
2.2.9 HBond_Lifetime.....	34
2.3 Fast Analysis Suite Summary.....	35
3. MHC Class 1 Protein Analysis.....	37
3.1 MHC Class 1 Proteins.....	37
3.2 Previous Literature.....	38
3.3 Simulation Details.....	39
3.4 Analysis.....	40
3.4.1 Residue Grouping Determination.....	41
3.4.2 Solvation.....	43
3.4.3 Apparent Diffusion Coefficient.....	44
3.4.3.1 1B0G.....	44
3.4.3.2 1EEZ.....	46
3.4.3.3 1IIF.....	48
3.4.3.4 All Diffusion.....	50
3.4.4 RDF.....	51
3.4.4.1 1B0G.....	51
3.4.4.2 1EEZ.....	53
3.4.4.3 1IIF.....	55
3.4.4.4 RDF Comparison for Key Domains.....	57
3.4.5 Hydrogen Bond Lifetime.....	60

## TABLE OF CONTENTS (continued)

Chapter	Page
3.4.5.1 1B0G.....	61
3.4.5.2 1EEZ.....	62
3.4.5.3 1I1F.....	63
3.4.5.4 Hydrogen Bond Lifetime Summary.....	64
3.4.6 Reorientation Times.....	66
3.4.6.1 1B0G.....	67
3.4.6.2 1EEZ.....	68
3.4.6.3 1I1F.....	69
3.4.7 MHC Summary.....	70
4. KRAS Diffusion Coefficient Analysis.....	71
4.1 KRAS Background.....	71
4.2 Mutations and Simulation Details.....	71
4.3 Residue Group Determination.....	73
4.4 Apparent Diffusion Coefficient.....	74
4.4.1 Wild Type KRAS.....	74
4.4.2 I36T and M67T Mutants.....	75
4.4.3 A59G Mutant (PDB ID 6ASE).....	77
4.4.4 A59G I36T and A59G M67T Double Mutants.....	78
4.4.5 Diffusion Coefficient Summary.....	80
5. Conclusions and Future Directions.....	82
5.1 Conclusions.....	82
REFERENCES.....	83

## LIST OF TABLES

Table	Page
Structural Information Output from Analysis Suite.....	42
Final Residue Grouping for Analysis.....	43
Diffusion Coefficients for 1B0G.....	44
Diffusion Coefficients for 1EEZ.....	46
Diffusion Coefficients for 1I1F.....	48
RDF Integrals for 1B0G.....	52
RDF Integrals for 1EEZ.....	54
RDF Integrals for 1I1F.....	56
Hydrogen Bond Lifetimes 1B0G.....	61
Hydrogen Bond Lifetimes 1EEZ.....	62
Hydrogen Bond Lifetimes 1I1F.....	63
Comparison of Hydrogen Bond Lifetimes.....	65
Reorientation Times 1B0G.....	67
Reorientation Times 1EEZ.....	68

LIST OF TABLES (continued)

Table	Page
Reorientation Times 111F.....	69
Diffusion Coefficients for Wild Type KRAS.....	75
Diffusion Coefficients for I36T Mutant.....	76
Diffusion Coefficients for M67T Mutant.....	77
Diffusion Coefficients for A59G Mutant.....	78
Diffusion Coefficients for A59G I36T Double Mutant.....	79
Diffusion Coefficients for A59G M67T Double Mutant.....	80
Diffusion Comparison all KRAS Structures.....	81

## LIST OF FIGURES

Figure	Page
Detailed Structural Analysis Application.....	6
Analysis Script Generator Application.....	12
Index_solvation_shell.sh.....	13
Solvation_shell_COM.sh.....	15
Index.sh.....	17
Msd.sh.....	19
Rdf.sh.....	23
RDF first solvation shell.....	25
Dipole.sh.....	27
Build_index_files.sh.....	30
Hbonding.sh.....	32
Hydration map 1B0G.....	45
Hydration map 1EEZ.....	47
Hydration map 1I1F.....	49

## LIST OF FIGURES (continued)

Figure	Page
RDF plot 1B0G.....	51
RDF plot 1EEZ.....	53
RDF plot 1I1F.....	55
RDF plots Alpha 1, Alpha 2, Peptide.....	57
RDF plot Alpha 3.....	58
Hydrogen Bond Lifetimes.....	65
Reorientation Diffusion Correlation 1B0G.....	67
Reorientation Diffusion Correlation 1EEZ.....	68
Reorientation Diffusion Correlation 1I1F.....	69
Diffusion Coefficient Comparison KRAS.....	81

## LIST OF ABBREVIATIONS

Å	Angstroms
PS	Picoseconds
NS	Nanoseconds
MHC	Major Histocompatibility Complex
GROMACS	GRoningen Machine for Chemical Simulation
MSD	Mean Squared Displacement
RDF	Radial Distribution Function
NVT	Canonical Ensemble moles (N) Volume (V) Temperature (T)
NPT	Isothermal-Isobaric Ensemble moles (N) Pressure (P) Temperature (T)

# CHAPTER 1

## INTRODUCTION

### 1.1 Hydration Shell Background

Water, the most ubiquitous solvent in biological systems, plays a crucial role in determining the physicochemical and functional landscapes of biomolecules, particularly proteins. While the properties of bulk water are well understood due to extensive studies, water molecules that directly interact with the protein surface, termed as the 'first solvation shell', exhibit unique behaviors. These molecules often form highly specific hydrogen bond patterns with the protein, serving not just as mere bystanders but as integral contributors to the protein's conformational flexibility, stability, and overall structure. Their dynamic interchange with the bulk solvent intricately impacts protein activities such as folding, binding, and catalysis, revealing insights into protein surface characteristics and accessibility. Thermodynamically, the reorganization or displacement of these water molecules during events like ligand binding or protein folding can significantly contribute to the observed enthalpy and entropy, and thereby the free energy of the processes. It has been shown by Fenimore, et al., that proteins' dynamics are directly tied with both bulk and hydration shell solvent. Instead of being semi-rigid, isolated entities, protein functions and movements are intricately linked to the larger solvent and their immediate hydration layer. They categorized protein motions into three distinct types: those influenced by the bulk solvent, those associated with the hydration layer, and inherent vibrational movements [1]. The movements driven by the solvent correspond with the dielectric fluctuations in the solvent and vanish in solid surroundings or in dehydrated proteins. These motions can be described by a Vogel–Tammann–Fulcher-like relationship and are associated with significant structural shifts, impacting processes like the entry and exit of ligands in myoglobin (Mb). On the other hand, movements related to the hydration shell align with rapid fluctuations in this

layer. They persist even when the protein is in a solid matrix but vanish in dehydrated states. Such movements predominantly influence side chains and facilitate events like the internal movement of ligands within Mb. This concept was expanded upon by Frauenfelder, et al, and explains the critical role of solvent in both protein folding and conformational changes needed for the release of CO or O<sub>2</sub> in myoglobin. During protein folding, the unfolded protein makes random walks towards the native state; along this random walk, many conformational substates happen until it reaches a transition state and ultimately the native state. Without the movement of solvent, this random walk would not occur, so the rate of folding becomes proportional to the rate coefficient of solvent. Because proteins which do not have similar structure have similar folding activation enthalpy, the model developed by McMahon can help explain why the activation enthalpy of protein folding is dominated by the solvent. They also apply this concept to the native state of myoglobin in which the rate constant for the release of CO or O<sub>2</sub> is given by equation 1 where  $k_{exit}$  is the rate constant for the release of CO or O<sub>2</sub>, the coefficient  $c$  represents the number of solvent fluctuations,  $k_{\alpha}$  is the rate constant of the  $\alpha$ -fluctuations, and  $n_{exit}$  is the number of fluctuations or elementary steps.

$$k_{exit}(T) = \frac{ck_{\alpha}(T)}{n_{exit}(T)} \quad (1)$$

Equation 1 shows that the enthalpic barrier which dictates ligand release is entirely due to the solvent [2]. Further exploration of the first hydration shell was explored by Fogarty, et al, specifically they looked at reorientation times in the coordination shell across several globular proteins. It is known from NMR studies that the hydration shell shows reorientation times 2-3 times slower than that of bulk water; Laage's research group was able to identify reasons for this retardation. Their simulations showed that there was a wide range of reorientation times across

the various hydration shell regions which they attribute to two different types of heterogeneity. Spatial heterogeneity represents the chemical nature of the residues in the region and topological features of the region (throughs, pockets, or protrusions) [3]. Dynamical heterogeneity explains how the flexibility of the specific protein region could affect the reorientation time, with the more flexible regions leading to faster reorientation. They showed that spatial heterogeneity was the main cause of the change in reorientation times and is not influenced by secondary structure or protein size [3]. Dynamical heterogeneity (conformational fluctuations) also has an effect, especially in concave regions where the hydration shell is confined.

Changes in solvents underscore the pivotal role they play as demonstrated in the case of myoglobin and protoheme by Beece, et al,. They found that over a range of solvent viscosity, the transition rates in heme-CO in protoheme and O<sub>2</sub> and CO in myoglobin are inversely proportional. This effect of solvent viscosity impacting reaction rates theory was applied by our research group to the hydration shell and specifically how the solvent layer affects protein dynamics. Simulations of CALB in the solvents water, n-butanol, tert-butanol, acetonitrile, and cyclohexane were performed to elucidate the connection between heterogeneous solvation shells, solvent, and protein flexibility. Four regions of interest which have large conformational changes were analyzed, and a viscosity-dependent trend was consistently shown. In this case, however, the local viscosity (related to diffusion rates of solvent at the surface), was considered. In the four organic solvents, increased local viscosity hindered regional flexibility and solvent dynamics. Dr. Dahanayake and Dr. Mitchell-Koch showed that there is a clear connection between solvent and regional dynamics and flexibility [4]. The exploration and understanding of the first solvation shell has garnered increasing attention within the scientific community. Recognizing this growing area of research, I have pioneered the development of a

comprehensive research tool. This innovative instrument, which I've named the "Fast Analysis Suite," is meticulously designed to facilitate the identification of solvation shells, delve deep into the chemical intricacies of each region, and simulate a host of both static and dynamic properties. By streamlining the analytical approach to hydration shells, the Fast Analysis Suite not only promotes efficiency but also ensures a reproducible methodology, setting a new standard for researchers worldwide.

## **1.2 The Need for a Reproducible and Fast Analysis Protocol**

Reproducibility is a cornerstone of scientific research. Without it, scientific findings remain unverified and their applicability questionable. Reproducible research allows for the confirmation of results by independent parties, enhancing the confidence in and the integrity of scientific discoveries. In the realm of protein analysis, where subtle changes in structural and dynamic parameters can influence biological function, the significance of reproducible research is even more pronounced. A fast and integrated analysis protocol is equally essential. Time is a precious commodity in the research landscape, where the swift transformation of data into knowledge can impact the progress of scientific endeavors significantly. A streamlined, efficient protocol can save countless hours of computational and researcher time, enabling faster data analysis and more immediate interpretation of results. Moreover, the integration of multiple analytical methods into a single suite facilitates a holistic understanding of the protein landscape. By providing an array of information from a single analysis, researchers can observe correlations and interactions between different protein characteristics that might otherwise be overlooked when each analysis is performed separately. A reproducible, fast, and integrated analysis protocol is not a luxury but a necessity in modern protein research. It enhances the reliability and efficiency of research, providing a comprehensive view of the protein landscape, and enabling

researchers to keep pace with the growing complexity of biological datasets. In light of this, the first part of this thesis details the development of a fast analysis suite designed to tackle these challenges head-on and provide a robust, efficient, and integrated platform for protein analysis. This suite is composed of two main applications that allow researchers to delve into protein structure and dynamics, revealing the hidden intricacies of proteins and their roles in biological systems. The second part of the thesis demonstrates the application of the Fast Analysis Suite by performing analysis on MHC Class 1 proteins and KRAS.

## CHAPTER 2

### OVERVIEW OF THE FAST ANALYSIS SUITE

#### 2.1 Structure Analysis Suite

This chapter will cover the main features of the Structure Analysis Suite which I have developed as a quick analysis tool for analysis of a broad range of structural features. When the user runs the program the GUI will open, and the user will have 3 analysis options to choose from SASA, Secondary Structure Analysis, and Detailed Residue Analysis; as shown in Figure 1.

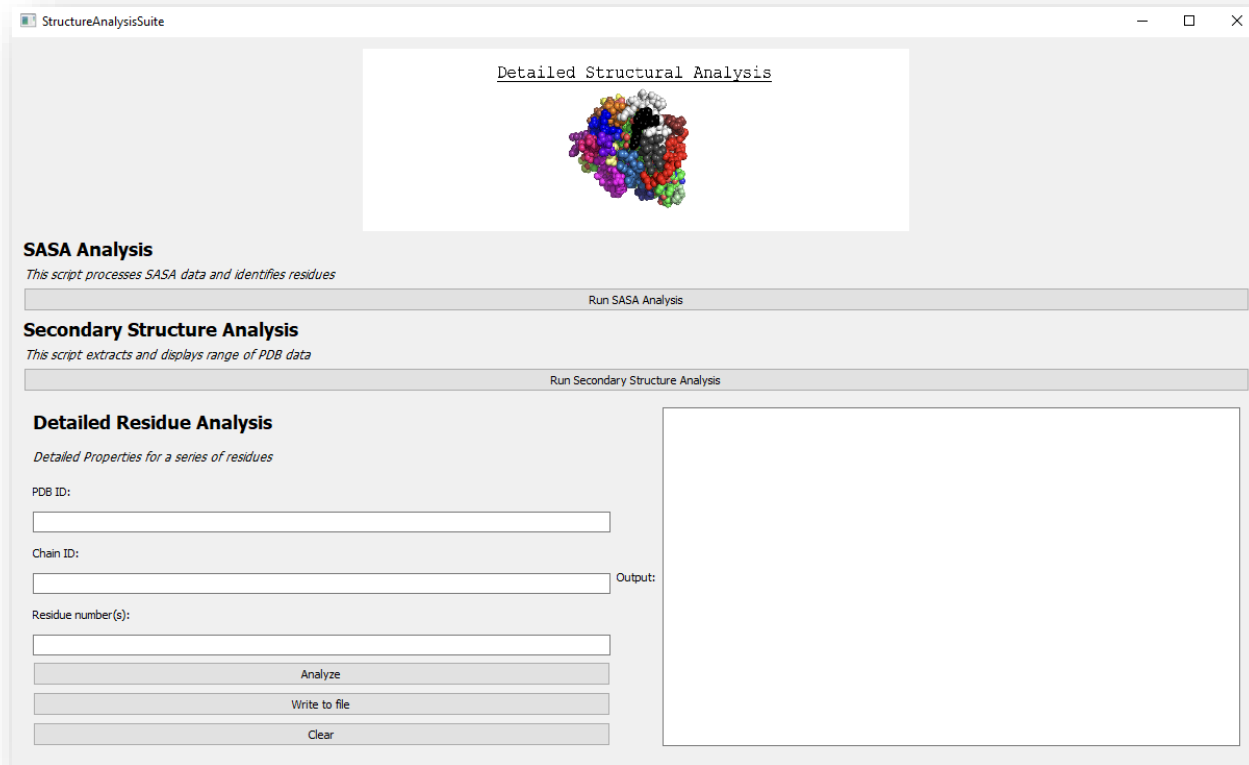


Figure 1: Detailed Structural Analysis Application

### 2.1.1 Solvent Accessible Surface Area

The solvent-accessible surface area (SASA) is a crucial parameter in the analysis of protein structure and function. SASA measures the surface area of a protein that is accessible to a solvent, providing key insights into the protein's interaction with its surrounding environment. SASA has important implications for understanding several aspects of protein behavior:

1. **Protein Folding and Stability:** The ratio of SASA to a protein's volume often correlates with the stability of the protein's structure. More stable, properly folded proteins tend to have lower SASAs as they pack their hydrophobic residues away from the solvent, while unstable or misfolded proteins often expose more of their surface area [5].
2. **Protein-Protein Interactions:** In protein-protein interactions, the interacting surfaces of the proteins are usually buried, decreasing the total SASA. Thus, changes in SASA can be indicative of protein-protein interactions, providing valuable information about binding sites and interaction dynamics.
3. **Protein-Ligand Binding:** SASA is also a key parameter in understanding protein-ligand binding. The binding of a ligand usually leads to a reduction in the protein's SASA, as the ligand occupies space that was previously accessible to the solvent.
4. **Enzymatic Activity:** For enzymes, the SASA can provide information about the active site. A pocket or groove in the enzyme that binds to the substrate often exhibits a unique SASA, indicative of its role in the enzyme's catalytic activity.
5. **Hydration Dynamics Analysis:** A detailed SASA for each residue of a protein is an essential starting point for identifying and sectioning the protein into regions for analysis.

Given its role in understanding protein stability, interactions, binding dynamics, and enzymatic activity, SASA is a fundamental parameter in protein analysis, making it a critical feature of the structure analysis suite developed in this study. There are many methods to measure SASA. For this suite I have chosen one developed by the University of Texas Medical Branch whose website for the analysis can be found at <https://curie.utmb.edu/getarea.html>. This website was chosen so that the user does not have to download any additional programs such as Chimera. Once at the website the user is prompted to load their pdb file and submit for analysis. The default probe radius of 1.4 nm is appropriate for pure water systems [6]. Once the user has saved the SASA data to a text file they are ready to load it into the Structure Analysis Suite. The python script reads the input file and groups the residues by a solvent accessible surface area of  $0.5 \text{ \AA}^2$  or greater. The output generates a list of continuous residues having SASA greater than  $0.5 \text{ \AA}^2$ , 1-12, 14-21, 25-30, etc. This information, in conjunction with secondary structure information, provides the user with enough information to appropriately divide their protein into sections of interest for analysis.

### **2.1.2 Secondary Structure Analysis**

The secondary structure of a protein refers to the local, regularly repeating structural motifs formed by amino acid residues in a polypeptide chain, including alpha-helices, beta-sheets, and turns or loops. These motifs arise from the hydrogen bonding between the peptide backbone atoms, which lend stability to the protein structure.

The secondary structure of a protein is vital for the following reasons:

1. **Structural Stability:** Alpha-helices and beta-sheets contribute to the stability of the protein due to the hydrogen bonding pattern within these structures. They form the core structural framework around which the protein folds.
2. **Functional Determination:** Certain secondary structures are characteristic of specific protein functions. For example, transmembrane proteins often contain alpha-helices, while enzymes may contain beta-sheets forming a catalytic barrel.
3. **Protein Folding:** The sequence of secondary structures in a protein forms a specific folding pattern which is crucial for the protein's tertiary structure and its function. Misfolding can lead to loss of function and, in some cases, cause disease states like Alzheimer's and Parkinson's [7].
4. **Interaction Sites:** Specific secondary structures can form interaction sites. For instance, alpha-helices often participate in protein-protein interactions or serve as ligand binding sites [8].

Considering these critical roles, understanding a protein's secondary structure is fundamental to comprehending its biological function, stability, and interaction with other molecules. Hydration dynamics analysis at residues of interest with key secondary structure features is vital to a greater understanding of environment at binding sites, mechanisms, and overall water behavior. Once the user clicks “run secondary structure analysis” they are prompted to load in a PDB file downloaded from RCSB.org. The PDB file from the RCSB contains secondary structure information. The python script reads the pdb file and extracts secondary structure data, and it

then outputs residue sequences starting with chain ID and either an alpha helix or beta sheet descriptor to the right.

### **2.1.3 Detailed Residue Analysis**

The detailed residue analysis section of the application provides in-depth information regarding a residue or residue sequence. Once the user provides PDB ID, Chain (A,B,C etc.) and residue number(s) and presses the analyze button, the output window to the right populates with detailed information about their sequence. It includes full amino acid name, 3 letter code, charge, hydrophobic or hydrophilic, whether it can form hydrogen bonds, total charge, % hydrophobic residues, and % that can form hydrogen bonds. The user has the option to write the information to a file, select more residues which will add the information to the output window, or clear the output window and start over. Overall, considering SASA, secondary structure, and detailed residue information allows researchers to focus on functionally relevant or structurally significant residues. The inclusion of these parameters in the fast analysis suite supports more effective utilization of GROMACS for protein analysis. This not only improves the efficiency of GROMACS simulations but also enhances the accuracy and interpretability of the results.

## **2.2 Analysis Script Generator**

With an ever-growing complexity of biological data, managing and analyzing this data in an efficient and reproducible manner becomes a critical aspect of research. The generation and execution of bash scripts play a significant role in the automation and streamlining of these processes, and having an efficient application to generate these scripts is of utmost importance. The analysis script generator successfully fulfills four essential aspects of successful research and data analysis.

1. **Efficiency:** Generating scripts manually can be a time-consuming process, particularly when dealing with multiple analyses or large datasets. An application that swiftly creates scripts significantly reduces the time spent on preliminary setup, enabling researchers to focus more on the actual analysis and interpretation of results.
2. **Reproducibility:** The reproducibility of results is a key criterion for their validity. An application that automates script generation ensures the same code is used every time an analysis is performed, enhancing the reproducibility of the results. It also facilitates the sharing of the exact methods used, as other researchers can run the same scripts to verify results.
3. **Error Reduction:** Manual coding is prone to errors, and even minor mistakes can lead to significant inaccuracies in results. Automating script generation minimizes the risk of such manual errors, ensuring the accuracy of the analysis. For example, a standard analysis comprises approximately thirteen separate series of residues. The script generator saves the user from having to input over 90,000 characters of code manually.
4. **Organization:** As the number of analyses increases, managing the various scripts can become challenging. By organizing scripts into dedicated folders like Diffusion, Dipole, Dipole\_Plots, HBond, HBond\_Lifetime, MSD, RDF, Residue\_Details, and Templates, the application enhances the ease of navigation, making it simpler for researchers to locate specific scripts and understand the sequence of analyses performed.

In light of these considerations, the fast analysis suite I have developed, with its application for efficient generation of bash scripts, significantly elevates the ease, reliability, and organization of protein and simulation analysis. This improved management of data and analyses paves the way for more focused and fruitful research, enabling the swift transformation of data into knowledge.

## 2.2.1 Navigating the Analysis Script Generator

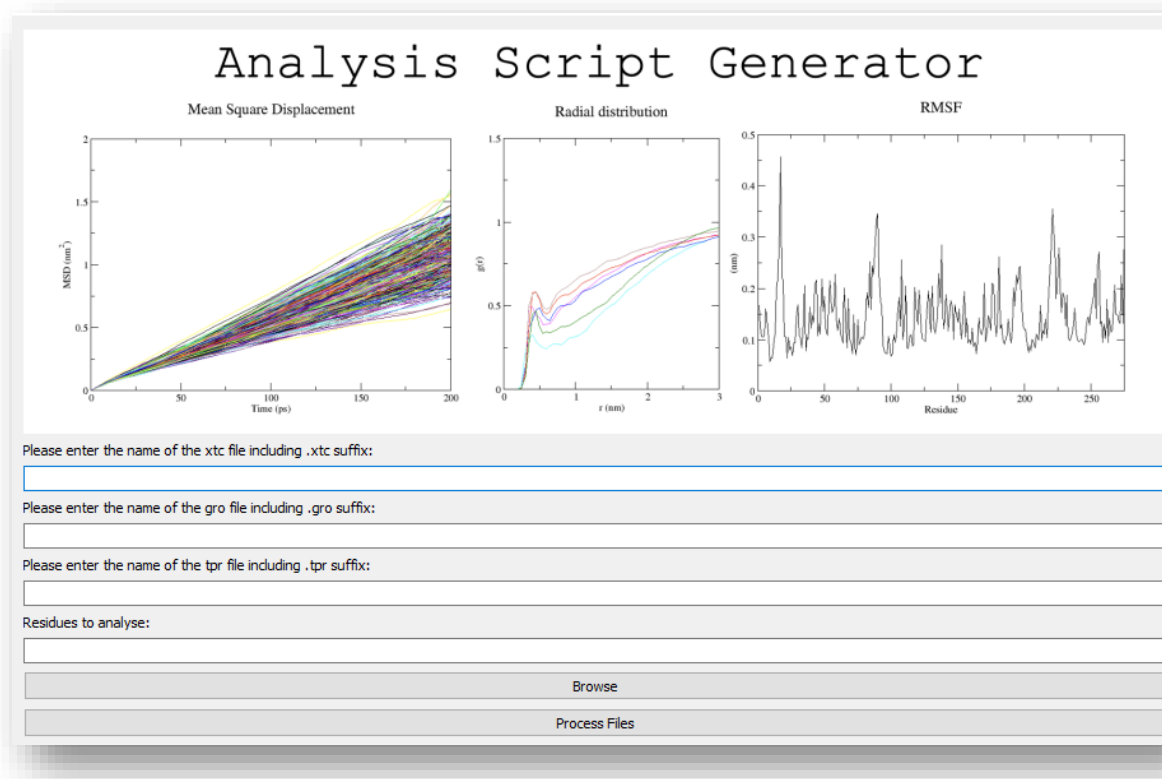


Figure 2: Analysis Script Generator Application

The Analysis Script Generator requires a total of four inputs from the user to generate and organize all analysis scripts. The first is the name of the GROMACS trajectory file which contains all the coordinates, velocities, forces, and energies for the simulation being analyzed. The second input is the name of the structure file used in the simulation which can be .gro or .pdb format. The third is a molecular topology file which contains all simulation parameters. The final input is the list of residues that the user wishes to analyze; the user is prompted to browse and select this file for the application to read. Once all inputs are provided and the user clicks the “Process Files” button scripts are generated and moved to one of ten folders: Diffusion, Dipole,

Dipole Plots, HBond, HBond Lifetime, MSD, RDF, Residue Details, Solvation, and Templates.

In this chapter I will provide details about each step in the analysis including theory, explanation of scripts, and why this analysis has been included in the Analysis Suite.

### 2.2.2 Solvation Folder

The Solvation Folder contains two scripts, `index_solvation_shell.sh` which creates index files for the residues of interest and `solvation_shell_COM.sh` which identifies the solvation shell. Figure 3 is an example of the `index_solvation_shell.sh` bash script which is running GROMACS's `gmx make_ndx` command to create index files for specific groups of atoms in a molecular system based on the provided `.gro` file, which is a GROMACS structure file containing the coordinate information of the system.

```
1 mkdir 1-25
2 cd 1-25

3 gmx make_ndx -f ../../../../1I7T.gro <<EOF
4 ri 1-25 & 8
5 name 18 1-25-sidechain
6 q
7 EOF
8 cd ..
```

Figure 3: `index_solvation_shell.sh`

1. **mkdir 1-25**: This line creates a new directory named "1-25".
2. **cd 1-25**: The script then navigates into the "1-25" directory.
3. **gmx make\_ndx -f ../../../../1I7T.gro <<EOF**: This command starts the execution of the GROMACS command `gmx make_ndx`, which creates an index file. The `-f` flag specifies

the input file, in this case, **1I7T.gro** located three directories above the current working directory.

4. **ri 1-25 & 8**: This command inside the EOF (end-of-file) markers is the input to **gmx make\_ndx** command. It selects the residues with indices from 1 to 25 (**ri 1-25**) and atoms that belong to group 8 (usually non-hydrogen atoms or sidechains, depending on the predefined groups in GROMACS based on individual systems). The **&** operator performs an intersection, i.e., it selects atoms that satisfy both conditions.
5. **name 18 1-25-sidechain**: This command assigns the name "1-25-sidechain" to group 18, which is the group formed by the previous command.
6. **q**: The **q** command quits the **gmx make\_ndx** command.
7. **cd ..**: The script navigates back to the parent directory.

The script repeats this for every residue sequence the user provides and outputs an `index.ndx` file which lists all atoms in the system, and creates a group named `1-25-sidechain` which lists all atoms associated with residues 1-25. Figure 4 is an example of the `solvation_shell_COM.sh` script which utilizes GROMACS `gmx select` command to select all waters within 6 Å; it takes a snapshot of the waters every 200 ps to ensure that the waters being observed are still within the first coordination shell and have not diffused out into bulk solvent. Figure 4 is an example of how the bash script is set up and a brief explanation of what the script is executing.

```
#!/bin/bash
1 cd 1-25
2 gmx select -f ../../1I7T_noPBC.xtc -s ../../1I7T_md3.gro -select "'Water" group SOL and within 0.6 of group "1-25-sidechain"' -seltype res_com -n index.ndx -dt 200 -oi <<EOF
3 EOF
4 cd ..
```

Figure 4: solvation\_shell\_COM.sh

1. **cd 1-25**: This command changes the current directory to the directory named "1-25".
2. **gmx select -f ../../1I7T\_noPBC.xtc -s ../../1I7T\_md3.gro -select "'Water" group SOL and within 0.6 of group "1-25-sidechain"' -seltype res\_com -n index.ndx -dt 200 -oi <<EOF**: This command runs the GROMACS **gmx select** command with several options:

- **-f ../../1I7T\_noPBC.xtc**: Specifies the input file as **1I7T\_noPBC.xtc**, a GROMACS trajectory file which contains the coordinates of the atoms at different times throughout the simulation.
- **-s ../../1I7T\_md3.gro**: Specifies the structure file **1I7T\_md3.gro**, another GROMACS file type that includes the system's atomic coordinates, velocities, and box size.
- **-select "'Water" group SOL and within 0.6 of group "1-25-sidechain"'**: This is the selection command that chooses atoms for further analysis. It selects water molecules (**group SOL**) that are within 0.6 nm of the group "1-25-sidechain".
- **-seltype res\_com**: This option specifies the type of selection. Here, **res\_com** means the selection is based on the center of mass of the residues.
- **-n index.ndx**: Specifies the index file **index.ndx** that contains the group "1-25-sidechain".

- **-dt 200**: This flag sets the time step for which to analyze the trajectory. It will read the trajectory from 0-200, 200-400, etc.
  - **-oi**: This option tells GROMACS to output the selected atoms' indices.
3. **EOF**: This pair of EOF (end-of-file) markers denotes the input to the **gmx select** command. However, since there is no input between the EOF markers, no input is provided to the command and this end-of-file command prompts the script to proceed.
  4. **cd ..**: The script navigates back to the parent directory.

This script redefines the solvation shell every 200 ps and writes the data to a file named `index.dat` which lists each water present at each 200 picosecond time interval. This file will be called to for diffusion analysis.

### 2.2.3 Diffusion Folder

The diffusion folder contains seven bash scripts which move and format index files, generate individual index files for water present in the solvation shell every 200 ps, and calculate mean squared displacement for each region of interest. The first `createdetails.sh` uses the `index.dat` file, which lists timestep, number of waters, and molecule number for each water. The `createdetails.sh` script copies the `index.dat` file and renames it `details.txt`. The next script `details.sh` uses the vi-editor command `:%s/^\{16\}//` which deletes the first sixteen columns in the `details.txt` file. The format of the `index.dat` file is set up in such a way that the first sixteen columns contain timestep information and number of waters, which need to be omitted for later scripts to read through and create index files from the list of water molecules. The next script is the `movefiles.sh` script which moves both `index.sh` and `msd.sh` scripts into each residue folder. The next script to run is the `bashindex.sh` script which moves into each residue folder and

executes the `index.sh` script. This will generate an `index.ndx` file for each 200 picosecond time interval listing waters present within that interval. Figure 5 is an example of the `index.sh` script followed by a brief explanation of the scripting.

```
#!/bin/bash
1 i=1
2 file="details.txt"
3 while read line
  do
4 Index_file="index${(i++)}.ndx"
5 gmx make_ndx -f ../../../../grofile -o $Index_file <<EOF
6 r $line
7 q
  EOF
8 rm -rf *#
9 done < details.txt
```

Figure 5: `index.sh`

1. **i=1**: This line initializes a variable **i** to 1. This variable will be used to number the output index files.
2. **file="details.txt"**: This line initializes a variable **file** to "**details.txt**". This is the file containing the residue numbers for which index files will be generated.
3. **while read line**: This command begins a while loop that reads the **details.txt** file line by line. Each line's contents are stored in the variable **line**.
4. **Index\_file="index\${(i++)}.ndx"**: This line creates a variable **Index\_file** that contains the name of the output index file. The index file is named "index1.ndx" for the first line of the text file, "index2.ndx" for the second line, and so on.

5. **gmx make\_ndx -f ../ ../1I7T\_md3.gro -o \$Index\_file <<EOF**: This command runs the GROMACS **gmx make\_ndx** command. The **-f** flag specifies the input **.gro** file, and the **-o** flag specifies the output index file's name.
6. **r \$line**: This command, within the EOF (end-of-file) markers, is the input to the **gmx make\_ndx** command. It selects the residue(s) specified by the current line of the text file.
7. **q**: The **q** command, also within the EOF markers, quits the **gmx make\_ndx** command.
8. **rm -rf \*#**: This line removes any files whose names end with "#". These are typically temporary files created during the execution of the script and are not needed after the script has finished running.
9. **done < details.txt**: This line ends the while loop and indicates that the loop should be executed once for each line in **details.txt**.

After this script has finished running the user is now ready to do mean squared displacement analysis. Mean Squared Displacement (MSD) is a statistical measure used to quantify the spatial extent of particle motion in a system. It's frequently used in studies of diffusion, especially in the field of physics, biophysics, and computational chemistry, and is defined as the average of the squared displacements of particles from their original position over a certain period of time. The mathematical expression is shown in equation 2 where  $r(t)$  represents the position of the particle at time  $t$ ,  $r(0)$  is the position at the initial time, and  $\langle \rangle$  denotes the average over all particles and time origins [9].

$$MSD = \langle (r(t) - r(0))^2 \rangle \quad (2)$$

In the case of simple diffusion, the MSD is proportional to time (linear time dependence) and the constant of proportionality is the diffusion coefficient,  $D$ , such that  $MSD = 2nDt$ , where  $n$  is the dimensionality of the diffusion. GROMACS calculates MSD using the `gmx msd` command. It computes the MSD as a function of the lag time, which is the time difference between the initial and final positions of particles. The command requires an input trajectory file, a topology file, and an index file (to specify the group of atoms for which to compute the MSD). In the context of molecular dynamics simulations, MSD can be useful for characterizing the diffusive behavior of atoms, molecules, or larger structures in the system, providing insights into system dynamics and interactions. When the `diffusion.sh` script is run, it moves into each residue folder and executes the `msd.sh` bash script. Figure 6 is an example and explanation of the `msd.sh` script.

```
#!/bin/bash
1 i=1 j=1
2 for k in `seq 0 200 100000`
do
3   First_frame="$k"
4   Last_frame=$((k+200))"
5   Index_file="index${(i++)}.ndx"
6   Output_file="msd${(j++)}.xvg"
7   group="18"
8   gmx msd -f ../../1B0G_noPBC.xtc -s ../../nvt_md.gro -o $Output_file -b $First_frame -e $Last_frame -n $Index_file -beginfit 10 -endfit 40 <<EOF
9   $group
10 EOF
11 rm -rf *#
done
```

Figure 6: `msd.sh`

1. The script defines two variables ‘**i**’ and ‘**j**’ and sets their initial value to 1.

2. The loop variable '**k**' represents the frames of the molecular dynamics simulation, with each frame representing a snapshot of the system at a particular point in time. The loop starts from 0 and increases in steps of 200 ps up to 100000 ps.
3. Within the loop, '**First\_frame**' is set to the current '**k**' value, and '**Last\_frame**' is set to '**k + 200**'. This sets a window of 200 frames over which the MSD is calculated.
4. '**Index\_file**' is the file where the output of the '**gmx\_msd**' command will be written. The filename changes with each iteration of the loop (incremented by '**j**').
5. '**Output\_file**' is the file where the output of the '**gmx\_msd**' command will be written. The filename changes with each iteration of the loop (incremented by '**j**').
6. The '**group**' variable is set to '**18**' which is the index group you created listing the solvent of interest within 6 Å of the solvent exposed residue sequence.
7. The script then runs the '**gmx\_msd**' command with a set of options:
  - '-f' specifies the trajectory file, which contains the position of the atoms over time.
  - '-s' specifies the structure file, which contains the initial configuration of the atoms.
  - '-o \$Output\_file' specifies where to write the output (within the same directory). The output files are written as msd1.xvg, msd2.xvg, etc.
  - '-b \$First\_frame -e \$Last\_frame' specifies the beginning and end of the time period over which to calculate the MSD.

- ‘-n \$Index\_file’ specifies the index file to use.
  - ‘-beginfit 10 -endfit 40’ are parameters related to a linear fit of the MSD as a function of time.
8. A Here Document (‘<<EOF ... EOF’) is used to supply the group number to the ‘**gmx msd**’ command. The number 18 is piped into the command as its input.
  9. The script then removes all files whose names end in ‘#’. The ‘-rf’ options to the ‘**rm**’ command indicate the removal should be recursive (including directories) and forceful (without asking confirmation).
  10. This loop will continue until all frames (0 to 100000 in increments of 200) have been processed.

After this script has finished within each residue folder there will be multiple msd.xvg plots. The number of plots is equal to simulation time divided by 200, for a 100,000-picosecond simulation each residue folder will have 500 msd.xvg plots.

#### 2.2.4 MSD Folder

This method of calculating Diffusion Coefficients uses the Einstein relation for Brownian motion in a fluid. The relationship between MSD and diffusion coefficients is given by the following equation.

$$MSD = 2 * n * D * t \quad (3)$$

Where MSD is the mean squared displacement, n is the number of dimensions (n=1 for 1D, n=2 for 2D, n=3 for 3D), D is the diffusion coefficient, and t is time. GROMACS simulations are a 3D system, and the slope is given by 2nD; we can calculate the diffusion coefficient by dividing

the slope by  $2n$ , or 6, in this 3D system. The first script to run is `move_msd.sh`, as this will make residue folders and move all plots to this folder. This consolidates all plots to a separate folder making the total file size smaller and easier to zip or transfer. After all plots have been moved, the user will run the `move_calc_regression.sh` bash script. This will move the `Linear_Regression.py` file into each folder and execute the python script. The python script loops over all msd files and calculates the diffusion coefficient for each one, converts to the appropriate units of  $\text{cm}^2/\text{s}$ , and saves the coefficients into a file named `Diff_Linear_Regression.txt`. The linear regression is performed over the most linear portion of the plot, between 10 and 40 ps. After the linear regression analysis has been performed the user will run the script `blockaveraging.sh` which executes the `block.averaging_20Blocks.py` script in each residue directory. The diffusion coefficients are broken up into 20 blocks representing 5 nanosecond time spans. When averaging in this manner, the user can better understand how diffusion fluctuates over time and potentially coinciding with structural changes. The `blockaveraging.sh` script produces a file named `Block_Averaging_20Blocks.txt` which lists the average for each block, the average of each block average, the standard deviation, and the variance. The user can then run the `get_diffusion_averages.sh` script which utilizes the “grep” command to extract the averages in each folder and write them to a file named `Diffusion_Averages.txt` which lists all diffusion coefficient information for all residue folders.

### **2.2.5 RDF Folder**

The Radial Distribution Function (RDF), also known as pair distribution function,  $g(r)$ , in a molecular system is a measure of the probability of finding a particle at a distance  $r$  from another particle, compared to that expected for a completely random distribution at the same density. It provides insights into the structural organization of systems at the atomic or molecular

level. It is important to note that the user needs to specify the groups of atoms for which they want to calculate the RDF when running in GROMACS. Internally, GROMACS calculates the RDF by creating a histogram of particle distances for each frame for the trajectory and then normalizing the histogram by the volume of the spherical shell corresponding to each distance bin and by the number density of the particles, so that in a completely homogeneous and isotropic system, the RDF approaches 1 at large distances. The user will run the `index_solvation_shell_rdf.sh` script which is the same as the script in the solvation folder; this script will generate the groups of atoms that GROMACS needs to calculate the RDF. After the index files have been generated the user is now ready to calculate the RDF by running the script `rdf.sh`. Below is an example of the script and a brief description of the code.

```
#!/bin/bash
1 cd 1-25
2 gmx rdf -f ../../1I7T_noPBC.xtc -s ../../1I7T_md3.gro -seltype res_com -n index.ndx -bin 0.05 <<EOF
3 18
4 EOF
```

Figure 7: `rdf.sh` script example

1. **cd 1-25**: This command changes the current directory to a directory named "1-25".
2. **gmx rdf -f ../../1I7T\_noPBC.xtc -s ../../1I7T\_md3.gro -seltype res\_com -n index.ndx -bin 0.05 <<EOF**: This command runs the GROMACS `gmx rdf` command with several options:
  - **-f ../../1I7T\_noPBC.xtc**: Specifies the input trajectory file as `1I7T_noPBC.xtc`.
  - **-s ../../1I7T\_md3.gro**: Specifies the structure file as `1I7T_md3.gro`.

- **-seltype res\_com**: Specifies that the selection of atoms should be based on the center of mass of residues.
  - **-n index.ndx**: Specifies the index file as **index.ndx**. This file contains the definition of atom groups used in the calculation which was generated with the `index_solvation_shell_rdf.sh` script.
  - **-bin 0.05**: Specifies the size of the bins to be used in the histogram for the RDF calculation. The unit is typically in nm.
3. **18** and **13**: These are the inputs to the **gmx rdf** command, provided between the EOF (end-of-file) markers. These numbers correspond to the group numbers defined in the **index.ndx** file. Group 18 represents the residues of interest created by `index_solvation_shell.sh` and group 13 represents water. The **gmx rdf** command calculates the RDF between these two groups of atoms.
4. **EOF**: This marker denotes the end of the input for the **gmx rdf** command.

After the RDF calculations have finished there will be a plot named `rdf.xvg` in each residue folder. The next analysis to perform is integrating the RDF plot. The RDF of pure water shows a distinct solvent shell out to 6 Å as shown in Figure 8. It is important for the user to identify where the solvation shell ends for proper integration. Non-water systems have distinct and unique solvation shells that may deviate from the 6 Å water solvation shell.

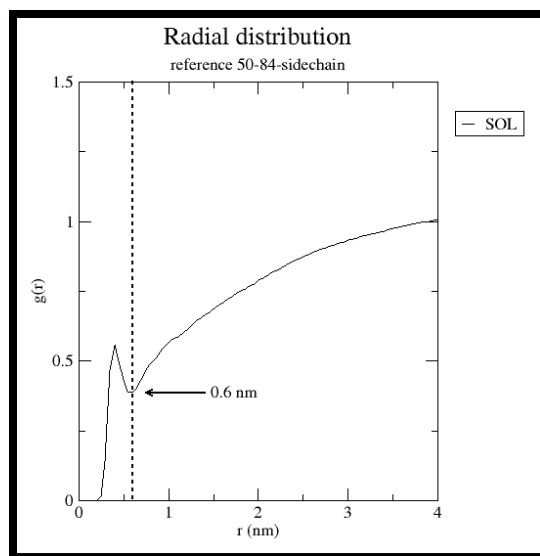


Figure 8: RDF first solvation shell

By integrating the RDF, one can obtain the cumulative number of neighboring atoms within a radius  $r$  from the reference atom(s). This provides us with information on the average number of nearest neighbors and allows us to identify the most probable distances between particles. For a liquid simulation, the first peak of the RDF corresponds to the first coordination shell, where the neighboring atoms are most densely packed. The integral of the RDF up to the first minimum gives the average number of atoms in the first coordination shell. To integrate the RDF plots, the user will run the script `rdf_integrate.sh`.

### 2.2.6 Dipole Folder

The dipole folder contains all scripts needed to calculate the dipole autocorrelation function using GROMACS. In molecular systems, the dipole moment is a measure of the overall polarity of the molecule, which depends on the spatial distribution of charges in the molecule. It is a vector quantity, meaning it has both a magnitude and a direction. The dipole moment can fluctuate over time, depending on the motion and interaction of the particles in the system. The

autocorrelation function is a measure of how a quantity (in this case, the dipole moment) correlates with itself as a function of time. It provides a measure of the time scale over which the dipole moment "forgets" its initial orientation, which is related to the rotational relaxation of the molecule. In GROMACS, the dipole autocorrelation function (ACF) is computed using the gmx dipoles module. It is a measure of how the dipole moment of a system changes over time and is a useful tool for understanding the dynamics of polar molecules, like water, in a system.

Mathematically, the dipole autocorrelation function,  $C(t)$ , is defined as follows:

$$C(t) = \langle M(0) \cdot M(t) \rangle \quad (4)$$

where  $M(0)$  and  $M(t)$  are the dipole moment vectors of the system at time zero and time  $t$ , respectively, and  $\langle \rangle$  denotes the time average. In GROMACS, the dipole ACF is computed by first calculating the total dipole moment vector of the system at each time step. This vector is a sum of the dipole moments of all the molecules in the system. Then, the autocorrelation function is computed as the average dot product of the dipole moment vectors at different time delays. This provides a measure of how the total dipole moment of the system changes over time. The resulting dipole ACF can then be used to extract useful information about the system. For example, the decay time of the ACF provides information about the rate at which the system's dipole moment relaxes back to its equilibrium value after being perturbed. This can provide insights into the system's dynamic behavior. The dipole ACF will be calculated every 200 ps, and the index.ndx files used in diffusion will also be used for the dipole autocorrelation function. The first script the user will run is the `move_index_diffusion_dipole.sh` bash script to copy and move the index.ndx files created in the diffusion folder to the dipole folder. These are the index files that list water within 6 Å of the protein every 200 ps. Next the user will run the

move\_dipolescript.sh script which will move the dipole.sh script into each residue folder. Once all scripts are in place the user can now run the dipole\_calc.sh script, which moves into each folder and executes the dipole.sh bash script. An example of the script is shown in Figure 9 followed by an explanation of the code.

```
#!/bin/bash
1 i=1 j=1
2 for k in `seq 0 200 100000`
do
3 First_frame="$k"
  Last_frame=$((k+200))
  Index_file="index${(i++)}.ndx"
  Output_file="dipcorr${(j++)}.xvg"
  group="18"
4   gmxdipoles -f ../.././xtcfile -s ../.././trjfile -c $Output_file -corr mol -P 1 -b $First_frame -e $Last_frame -n $Index_file <<EOF
  $group
  EOF
  rm -rf *#
done
```

Figure 9: dipole.sh script

1. The script sets initial values for the variables **i** and **j** to 1.
2. A loop is then started that runs from 0 to 100000 in steps of 200. The loop variable **k** represents the frames of the molecular dynamics simulation, with each frame representing a snapshot of the system at a certain point in time.
3. Within each iteration of the loop:
  - **First\_frame** and **Last\_frame** are defined as **k** and **k + 200** respectively, indicating a window of 200 frames over which the dipole moment is calculated.

- **Index\_file** is defined as a file containing indices of the molecules for which the dipole moment should be calculated. Its name changes with each iteration of the loop.
- **Output\_file** is defined as the file where the output of the **gmx dipoles** command will be written. Its name also changes with each iteration of the loop.
- **group** is set to "18", indicating that the 18th group defined in the index file will be analyzed. This is the group containing a snapshot of all waters within 6 Å of the residues every 200 ps.

4. The **gmx dipoles** command is then executed with several options:

- **-f ../../xtcfile** specifies the input trajectory file.
- **-s ../../tprfile** specifies the input structure file.
- **-c \$Output\_file** specifies the output file for the total and/or average cosine of the angle with the z-axis.
- **-corr mol** specifies that the correlation function should be calculated per molecule.
- **-P 1** means that only the first principal component of the inertia tensor is used.
- **-b \$First\_frame -e \$Last\_frame** specifies the start and end time for analysis in the trajectory file.
- **-n \$Index\_file** specifies the index file to be used.

5. The **gmx dipoles** command is run for the group of molecules specified in the **Index\_file** within the given **First\_frame** and **Last\_frame** time window.
6. This process is repeated for every 200 ps-frame window in the specified frame range.

### 2.2.7 Dipole\_Plots Folder

After running the `dipole_calc.sh` script there will be 500 `dipcorr.xvg` plots in each of the residue folders. The next step for analysis is to do curve fitting for the ACF function. The bash script `move_dipole_plots.sh` will move all dipole autocorrelation plots from the Dipole folder to the Dipole\_Plots folder. After all plots have been moved, the user will then run the script `dipole_averaging.sh`, this bash script calls to the dipole averaging python script which averages all plots and generates a new plot named `average_dipcorr.xvg`. Next is to fit the curve and get the optimized coefficients for the fitted equation. To do this the user will run the script `curve_fitting.sh`, this script moves the curve fitting python script into each folder and executes it, producing a plot and optimized parameters in each folder. After the curve fitting, running the bash script `get_parameters.sh` will generate a file named `Optimized_Coefficients.txt` with all coefficients listed for each residue folder.

### 2.2.8 HBond Folder

The HBond Folder contains 8 scripts which are used to move and create index files and calculate the hydrogen bond autocorrelation function. First the user must move `index.ndx` files from both the solvation and diffusion folders, the index file from solvation creates the group defining the atoms of the residue of interest. The index files from diffusion create a unique group that represents all waters within 6 Å every 200 ps. All index files can be moved by running both `move_index_diffusion_hbond.sh` and `move_index_solvation_hbond.sh`. Now that

the 500 index files from diffusion and the index file from solvation are in each residue folder, several new index files which combine the two needs to be made. To create new index files, the user must first run `move_ndx_hbond.sh` to move all necessary bash scripts into each folder. After appropriate scripts have been moved, the user can run the script `build_index_files.sh` which will consolidate the residues of interest and the waters of interest into a series of index files. Figure 10 is an example of the script that accomplishes this with a brief description of the code.

```
#!/bin/bash
1 i=1 j=1
2 for k in `seq 0 200 100000`
   do
3     First_frame="$k"
4     Last_frame="$((k+200))"
5     Index_file="index$((i++)).ndx"
6     Output_file="res_water$((j++)).ndx"
7     gmx make_ndx -f ../.././1I7T_md3.gro -o $Output_file -n index.ndx $Index_file <<EOF
8     EOF
9     rm -rf *#
10 done
```

Figure 10: `build_index_files.sh` example

1. **i=1 j=1**: This line initializes two variables **i** and **j** to 1. These variables will be used to number the output index files.
2. **for k in seq 0 200 100000**: This begins a loop that iterates over a sequence of numbers from 0 to 100000 in steps of 200. Each number in the sequence is assigned to the variable **k**.
3. **First\_frame="\$k"** and **Last\_frame="\$((k+200))"**: These lines assign the current value of **k** and **k+200** to the variables **First\_frame** and **Last\_frame**, respectively. These variables represent the range of frames to be analyzed.

4. **Index\_file="index\$(i++).ndx"** and **Output\_file="res\_water\$(j++).ndx"**: These lines create variables **Index\_file** and **Output\_file** that contain the names of the input and output index files, respectively. The names are numbered according to the current iteration of the loop.
5. **gmx make\_ndx -f ../../1I7T\_md3.gro -o \$Output\_file -n index.ndx \$Index\_file <<EOF**: This command runs the GROMACS **gmx make\_ndx** command. The **-f** flag specifies the input **.gro** file, **-o** flag specifies the output index file's name, and **-n** flag indicates an existing index file to add the group to.
6. **q**: The **q** command, within the EOF (end-of-file) markers, quits the **gmx make\_ndx** command.
7. **rm -rf \*#**: This line removes any files whose names end with "#". These are typically temporary files created during the execution of the script and are not needed after the script has finished running.
8. **done**: This line ends the loop.

After the new index files have been made the hydrogen bond analysis can be run. The script `run_hbonding.sh` will move the `hbonding.sh` script into each residue folder and execute the `hbond` command in GROMACS which calculates the number of hydrogen bonds between water in the first solvation shell and the residues of interest over the trajectory and calculate the hydrogen bond autocorrelation function. Figure 11 is an example of the script and a brief explanation of what it is doing.

```

#!/bin/bash
1 i=1 j=1 l=1
2 for k in `seq 0 200 100000`
do
3 First_frame="$k"
  Last_frame="$((k+200))"
4 Index_file="res_water$((i++)).ndx"
  Output_file1="numHbonds$((j++)).xvg"
  Output_file2="lifeH$((l++)).xvg"
5     gmx hbond -f ../../xtcfile -s ../../tprfile -num $Output_file1 -life $Output_file2 -b $First_frame -e $Last_frame -n $Index_file -ac <<EOF
6     18
7     37
8     EOF
7 rm -rf *#
8 done

```

Figure 10: hbonding.sh script

1. **i=1 j=1 l=1**: This line initializes three variables **i**, **j**, and **l** to 1. These variables are used to increment the filename numbers for each run of the loop.
2. **for k in seq 0 200 100000`**: This starts a loop that iterates over a sequence of numbers from 0 to 100000, with a step of 200. Each number in the sequence is assigned to the variable **k**.
3. **First\_frame="\$k"** and **Last\_frame="\$((k+200))"**: These lines assign the current value of **k** and **k+200** to the variables **First\_frame** and **Last\_frame**, respectively. These variables represent the range of frames to be analyzed in each loop iteration.
4. **Index\_file="res\_water\$((i++)).ndx"**, **Output\_file1="numHbonds\$((j++)).xvg"**, and **Output\_file2="hbac\$((l++)).xvg"**: These lines create variables **Index\_file**, **Output\_file1**, and **Output\_file2** that contain the names of the input index file and the output files for the number of hydrogen bonds and the autocorrelation function, respectively. The names are numbered according to the current iteration of the loop.

5. **gmx hbond -f ../../xtcfile -s ../../tprfile -num \$Output\_file1 -b \$First\_frame -e \$Last\_frame -n \$Index\_file -P 1 -temp 300 -ac \$Output\_file2 <<EOF**: This command runs the GROMACS **gmx hbond** command, which calculates the number of hydrogen bonds and the autocorrelation function for the hydrogen bonds.
  - **-f ../../xtcfile** specifies the input trajectory file, and **-s ../../tprfile** specifies the input topology file.
  - **-num \$Output\_file1** specifies the output file for the number of hydrogen bonds.
  - **-b \$First\_frame -e \$Last\_frame** specifies the beginning and end frames for the analysis.
  - **-n \$Index\_file** specifies the index file that contains the groups of atoms for the analysis.
  - **-P 1** sets the periodicity for the autocorrelation function.
  - **-temp 300** sets the temperature for the calculation in Kelvin.
  - **-ac \$Output\_file2** specifies the output file for the autocorrelation function.
6. **18** and **37**: These numbers are the input to the **gmx hbond** command, provided between the EOF (end-of-file) markers. They correspond to the group numbers defined in the **index.ndx** file. The **gmx hbond** command calculates the hydrogen bonds and their autocorrelation function between these two groups of atoms.
7. **rm -rf \*#**: This line removes any temporary files created during the execution of the script that end with "#".
8. **done**: This ends the loop.

### 2.2.9 HBond\_Lifetime

In GROMACS, the autocorrelation function (ACF) of hydrogen bonds can be calculated using the `gmx hbond` module. This module first identifies hydrogen bonds in the system based on geometric criteria: a maximum donor-acceptor distance and a minimum angle between donor-hydrogen-acceptor. The ACF of the hydrogen bonds is then calculated, which describes the likelihood that if a hydrogen bond exists at a certain time, it will still exist after a certain delay time. Mathematically, the autocorrelation function,  $C(t)$ , can be defined as:

$$C(t) = \frac{\langle h(0)h(t) \rangle}{\langle h(0)h(0) \rangle} \quad (5)$$

where  $h(t)$  is the existence of a hydrogen bond at time  $t$  (1 if it exists, 0 if it does not) and  $\langle \rangle$  denotes the time average. The decay of this autocorrelation function gives information about the lifetime of the hydrogen bonds. A rapid decay would indicate short-lived hydrogen bonds, while a slow decay indicates long-lived hydrogen bonds. The exponential fit is a commonly used method to analyze the decay of the ACF. It has the form:

$$C(t) = A1 * \exp\left(-\frac{t}{\tau_1}\right) + A2 * \exp\left(-\frac{t}{\tau_2}\right) \quad (6)$$

where  $A$  is a normalization constant,  $t$  is the time, and  $\tau$  is the characteristic time, or in this case, the hydrogen bond lifetime. By fitting the calculated ACF to this equation, one can extract the value of  $\tau$ , which gives the average lifetime of the hydrogen bonds in the system. The `HBond_Lifetime` folder contains 4 scripts which will be utilized to calculate hydrogen bond lifetime by doing a multi exponential curve fitting of the hydrogen bond ACF plots previously generated. By running the script `move_hb_data.sh` all autocorrelation plots and the output “screen.txt” will be moved to the `HBond_Lifetime` folder. After all plots have been moved the

user will run the script `hbond_averaging.sh` which will average all plots over the entire trajectory producing a new plot named `average_HBond.xvg`. The next step is to perform a multi exponential curve fitting, this is done by running the script `curve_fitting_hbond.sh` which will generate a text file listing all coefficients and the calculated average hydrogen bond lifetime using the equation:

$$\textit{Average hydrogen bond lifetime} = (A1 * \tau1) + (A2 * \tau2) \quad (7)$$

This lifetime provides valuable insight into the stability of the hydrogen bond network in the system and can be especially useful when studying water dynamics in relation to protein structures. It's important to note that hydrogen bond lifetime gives the time during which a hydrogen bond remains without breaking once formed. This can provide key insights into how hydrogen bonding contributes to the structure and dynamics of the system. In the context of the MHC Class 1 proteins, understanding hydrogen bond lifetimes could provide key insights into protein conformational stability and interactions with other molecules.

### **2.3 Fast Analysis Suite Summary**

Utilizing this advanced fast analysis suite, the structure and dynamics of hydration layers have been effectively characterized. This is achieved through examining radial distribution functions (RDF), diffusion rates, reorientation times, and hydrogen-bond lifetimes. Such detailed analysis surpasses what can be achieved with spectroscopic methods, which either offer average global properties through THz-to-GHz spectroscopy and neutron diffraction or limited site-specific details via 2D-IR, ODNP, or time-dependent fluorescence. Previously, conducting such comprehensive molecular dynamics (MD) simulations necessitated custom scripts and expert handling, a task undertaken by only a few specialized groups globally. However, this Fast

Analysis Suite has revolutionized the process, creating 55 scripts and over 6,000 lines of code, transforming a complex and error-prone task into a swift and efficient novel analysis method.

## CHAPTER 3

### MHC CLASS I PROTEIN ANALYSIS

#### 3.1 MHC Class I Proteins

Major Histocompatibility Complex (MHC) Class I proteins are essential components of the immune system and play a crucial role in immune response regulation. MHC Class I molecules are present on the surface of almost all nucleated cells, serving as a critical interface between the adaptive immune system and the cellular microenvironment. MHC Class I proteins are heterodimeric, comprising a polymorphic heavy chain (the  $\alpha$ -chain) non-covalently associated with a light chain (the  $\beta$ 2-microglobulin). The  $\alpha$ -chain, further divided into three domains ( $\alpha$ 1,  $\alpha$ 2,  $\alpha$ 3), associates with  $\beta$ 2-microglobulin to form the final MHC Class I complex. The  $\alpha$ 1 and  $\alpha$ 2 domains form a peptide-binding groove, which displays endogenously derived peptides to cytotoxic T lymphocytes (CTLs) [10]. These peptides are typically derived from intracellular pathogens such as viruses, effectively marking infected cells for destruction by CTLs. Conversely, the  $\alpha$ 3 domain and  $\beta$ 2-microglobulin interact with the T cell receptor (TCR) and CD8 co-receptor on CTLs, respectively, providing stability and specificity to the interaction. It is worth noting that the  $\beta$ 2-microglobulin is non-polymorphic, thereby serving as a conserved feature among MHC Class I proteins across different individuals. The conservation of chains A and B, also known as the  $\alpha$ 1 and  $\alpha$ 2 domains, in MHC Class I proteins is of paramount importance. While these regions do exhibit variability, certain aspects remain highly conserved, ensuring the functionality of these proteins. They contribute to the formation of the peptide-binding groove, and their conservation ensures that MHC Class I molecules can present a wide array of peptides to CTLs. This conservation has implications for the stability of MHC Class I molecules and their ability to interact effectively with the antigen processing machinery. Any variation in the conserved regions could potentially lead to structural anomalies, compromising

the binding of peptides and the recognition by TCR, and thus impairing the immune response. In the context of the present study, understanding the structural and functional nuances of MHC Class I proteins forms the basis for Molecular Dynamics (MD) simulations.

### **3.2 Previous Literature**

The exploration of how peptides influence the behavior of Major Histocompatibility Complex (MHC) proteins has been a subject of extensive research and inquiry. Notably, the work of Christian Freund and colleagues has introduced a compelling hypothesis. They postulate that the specific bound peptide exerts an influence on the flexibility of the binding region of MHC proteins. This, in turn, leads to an increased reliance on tapasin, a chaperone protein that aids in stabilizing the MHC protein-peptide complex during the assembly process [11]. Additionally, the investigations conducted by Baker and his team have contributed to our understanding of this intricate relationship. Their studies have suggested the existence of peptide-dependent motions within MHC proteins. Furthermore, these peptide-protein interactions have been shown to impact the free energy landscape of the system, resulting in correlated changes in protein dynamics [12]. These findings underscore the complex interplay between peptides and MHC proteins, highlighting the far-reaching consequences of peptide variations on protein behavior. Building upon these studies, there emerges a compelling question: how do peptide variants affect the solvation shell surrounding MHC proteins? It is well-established that alterations in protein dynamics often lead to changes in hydration dynamics. Consequently, this research embarks on a comprehensive exploration of the hydration shell surrounding three distinct MHC proteins. The overarching objective is to discern whether hydration dynamics remain conserved or undergo alterations in response to peptide variants. This investigation represents a critical step toward

unraveling the intricate relationship between peptides, protein dynamics, and hydration properties within the context of MHC proteins.

### **3.3 Simulation Details**

The amber03 forcefield was employed for these simulations. The choice of water model is an essential aspect of molecular dynamics simulations because it significantly influences the behavior of the solute. In our study, we opted for the SPC/E water model. This extended simple point charge model is known for closely matching the experimental diffusion properties of water, making it an optimal choice for our study focusing on water diffusion [13]. The initial 3D structures for the MHC Class 1 proteins were sourced from the RCSB Protein Data Bank. All the water molecules from the crystal structure were removed, ensuring a clean starting point for the simulations. Each protein was then positioned in a cubic simulation box with a minimum distance of 1.2 nm from the protein to the box boundaries. This setup prevented the protein from interacting with its periodic images and provided enough room for solvation. Following the placement of the proteins, the simulation box was filled with the water molecules from the SPC/E water model. Seven sodium counterions were added to each system to neutralize the overall charge. This step created a balanced environment, similar to physiological ionic conditions, for the accurate simulation of the system's behavior. The energy minimization step utilized the steepest descent algorithm to relieve any strain from the system initialization and lead the system toward a local potential energy minimum. This critical step helped eliminate any unfavorable or unrealistic interactions before the dynamic calculations. After energy minimization, an NVT equilibration was carried out for 1000 ps with position restraints on the protein's heavy atoms. This stage allowed the solvent to relax around the solute and achieve a stable temperature. This was followed by an NPT equilibration for 1000 ps, allowing the system

to reach a stable pressure and density. With the system well-equilibrated, we entered the production phase, releasing all position restraints for an unconstrained simulation under NVT conditions. Each production run was carried out over 100 ns, effectively capturing the temporal evolution of the system. To enhance the reliability and reproducibility of our findings, each production run was repeated three times. This resulted in an ensemble of dynamic trajectories, providing a comprehensive and reliable basis for our analysis. By combining these carefully chosen components, we were able to build a robust simulation framework capable of revealing the complex molecular behaviors of the MHC Class 1 proteins under study.

### **3.4 Analysis**

The Fast Analysis Suite played a crucial role in all structure analysis and script generation processes. This innovative tool proved to be not only efficient but also remarkably reliable, having been employed nine times in total - corresponding to each of the nine simulation runs performed. Across all these runs, the script generation tool consistently provided error-free results and maintained absolute consistency in its output. Its versatility and robustness underscore the value of the tool as a linchpin in the computational and simulation procedures. Moving onto the subsequent sections of this chapter, a more in-depth discussion and exploration of the suite's application and performance is provided. Each method of analysis, as mentioned and briefly introduced in Chapter Two will be demonstrated, utilized, and interpreted in the context of our comprehensive simulation analysis. Through a step-by-step unfolding of each method, we aim to elucidate how these individual analyses come together to form a cohesive, holistic understanding of the system under study. The goal of this thesis is to provide not only a testament to the effectiveness and capabilities of the Fast Analysis Suite but also to illustrate how these advanced computational tools can be harnessed to generate precise, reliable data for

complex molecular dynamics simulations. By shedding light on the practical application of the suite, we strive to highlight its potential as a valuable tool in the wider field of computational biology, particularly in studies involving protein analysis and simulation.

### **3.4.1 Residue Grouping Determination**

Deciphering the essential clusters of residues was significantly facilitated by the employment of the Structure Analysis Suite. This software suite yielded valuable insights into the residues contributing to the secondary structure, as well as those significantly influencing Solvent Accessible Surface Area (SASA). By analyzing the secondary structure, we were able to understand the protein's architecture on a broader level, while the SASA provided a measure of the protein surface exposed to the solvent, both of which are essential aspects in the study of protein dynamics and functionality. Upon analysis with the Structure Analysis Suite, the data, along with additional information from <https://curie.utmb.edu/getarea.html> was integrated into the Analysis Suite for comprehensive grouping of residues. By coupling the secondary structure information with the SASA data, we were able to conduct a more strategic grouping, narrowing down our focus to the residues playing vital roles in the protein's overall structure and behavior. In Table 1, the output derived from this grouping process for each protein is shown. The output files represent residues which are part of secondary structures and exhibit a solvent accessible surface area greater than  $0.5 \text{ \AA}^2$ . These residues, by virtue of their exposure to the solvent, present key interaction points for the dynamics of water molecules around the protein. All secondary structures that were exposed to the solvent were then collated into specific groups for focused and detailed analysis of the water dynamics surrounding these pivotal features. This tactical grouping allowed us to dive deeper into the intricacies of water behavior in the vicinity of these critical protein regions. Table 2 illustrates the final set of residue groups that were chosen for

further analysis. This strategic assortment of residues, carved out from a pool of hundreds, was the fruit of our rigorous grouping process and served as the key focus points for our subsequent investigations into the water dynamics around the MHC Class 1 proteins.

Table 1: Structural Information output from Suite

Secondary Structure	SASA	Secondary Structure	SASA	Secondary Structure	SASA
1B0G	1B0G	1EEZ	1EEZ	1IIF	1IIF
A 3-12 SHEET	1-4 A	A 3-12 SHEET	1-4 A	A 3-12 SHEET	1-8 A
A 21-28 SHEET	6-8 A	A 21-28 SHEET	6-8 A	A 21-28 SHEET	10-23 A
A 25-227 HELIX	12-23 A	A 25-228 HELIX	10-25 A	A 25-227 HELIX	29-32 A
A 31-37 SHEET	29-32 A	A 31-37 SHEET	29-32 A	A 31-37 SHEET	34-51 A
A 38-149 HELIX	35-66 A	A 37-150 HELIX	34-66 A	A 38-149 HELIX	53-66 A
A 50-54 HELIX	68-69 A	A 46-47 SHEET	68-69 A	A 52-161 HELIX	68-69 A
A 52-161 HELIX	71-73 A	A 49-54 HELIX	71-73 A	A 54-256 HELIX	71-73 A
A 54-256 HELIX	75-76 A	A 51-162 HELIX	75-76 A	A 57-84 HELIX	75-80 A
A 57-84 HELIX	78-80 A	A 53-255 HELIX	78-80 A	A 63-174 HELIX	82-94 A
A 63-174 HELIX	82-94 A	A 56-85 HELIX	82-94 A	A 76-179 HELIX	96-100 A
A 76-179 HELIX	104-113 A	A 62-175 HELIX	97-99 A	A 94-103 SHEET	104-116 A
A 94-103 SHEET	118-139 A	A 75-180 HELIX	104-115 A	A 109-118 SHEET	118-139 A
A 109-118 SHEET	141-142 A	A 94-103 SHEET	118-139 A	A 121-126 SHEET	141-142 A
A 121-126 SHEET	144-155 A	A 109-118 SHEET	141-142 A	A 133-135 SHEET	144-159 A
A 133-135 SHEET	157-159 A	A 121-126 SHEET	144-159 A	A 186-195 SHEET	161-163 A
A 186-193 SHEET	161-163 A	A 133-135 SHEET	161-163 A	A 198-206 SHEET	165-167 A
A 198-208 SHEET	165-167 A	A 186-192 SHEET	165-167 A	A 214-219 SHEET	169-170 A
A 214-219 SHEET	169-171 A	A 186-192 SHEET	169-170 A	A 228-230 SHEET	173-200 A
A 241-250 SHEET	173-200 A	A 198-208 SHEET	173-174 A	A 243-249 SHEET	206-207 A
A 257-262 SHEET	206-207 A	A 198-208 SHEET	176-200 A	A 257-262 SHEET	209-244 A
A 270-273 SHEET	209-216 A	A 214-219 SHEET	206-207 A	A 270-273 SHEET	246-248 A
B 6-11 SHEET	218-244 A	A 222-223 SHEET	209-244 A	B 6-11 SHEET	250-258 A
B 20-30 SHEET	246-248 A	A 229-230 SHEET	247-258 A	B 20-28 SHEET	262-275 A
B 36-41 SHEET	250-258 A	A 234-235 SHEET	262-275 A	B 35-41 SHEET	0-22 B
B 62-71 SHEET	262-275 A	A 241-250 SHEET	0-22 B	B 64-71 SHEET	28-29 B
B 78-83 SHEET	0-22 B	A 241-250 SHEET	28-29 B	B 78-84 SHEET	31-38 B
B 91-94 SHEET	28-29 B	A 257-262 SHEET	31-38 B	B 91-94 SHEET	40-54 B
	31-54 B	A 270-272 SHEET	40-54 B		56-61 B
	56-61 B	B 6-11 SHEET	56-61 B		63-71 B
	63-79 B	B 6-11 SHEET	63-71 B		73-79 B
	83-99 B	B 21-30 SHEET	73-79 B		83-99 B
	1-9 C	B 21-30 SHEET	83-99 B		1-9 C
		B 36-41 SHEET	1-9 C		
		B 44-45 SHEET			
		B 50-51 SHEET			
		B 55-56 SHEET			
		B 62-70 SHEET			
		B 62-70 SHEET			
		B 78-83 SHEET			
		B 91-94 SHEET			

Table 2: Final Residue Grouping for Analysis

Residues For Analysis		
1B0G	1EEZ	111F
1-12	1-12	1-12
13-23	13-25	13-23
29-32	29-32	29-32
35-49	34-48	34-48
50-84	49-85	49-84
85-94	86-94	85-94
104-113	104-115	104-116
118-137	118-136	118-137
138-149	137-180	138-179
152-179	181-200	180-200
180-200	206-224	206-224
206-224	225-228	225-227
225-227	229-244	228-244
228-244	247-252	246-253
246-253	253-255	254-255
254-256	256-258	256-258
262-275	262-275	262-275
276-298	276-298	276-298
307-330	307-330	307-330
332-337	332-337	332-337
339-355	339-355	339-355
359-375	359-375	359-375
376-384	376-384	376-384

Text files listing residues in Table 2 were imported into the Analysis Script generator to create all bash scripts used for analysis. This was done for all 9 trajectories.

### 3.4.2 Solvation

The scripts created in the Solvation folder were utilized to define the solvation shell based on the topological and sequence information obtained from the structure analysis application. After the solvation shell has been defined waters present within 6 Å were recorded in 200 ps increments. This definition of the solvation shell is utilized for diffusion analysis, reorientation time determination, and hydrogen bond lifetime analysis.

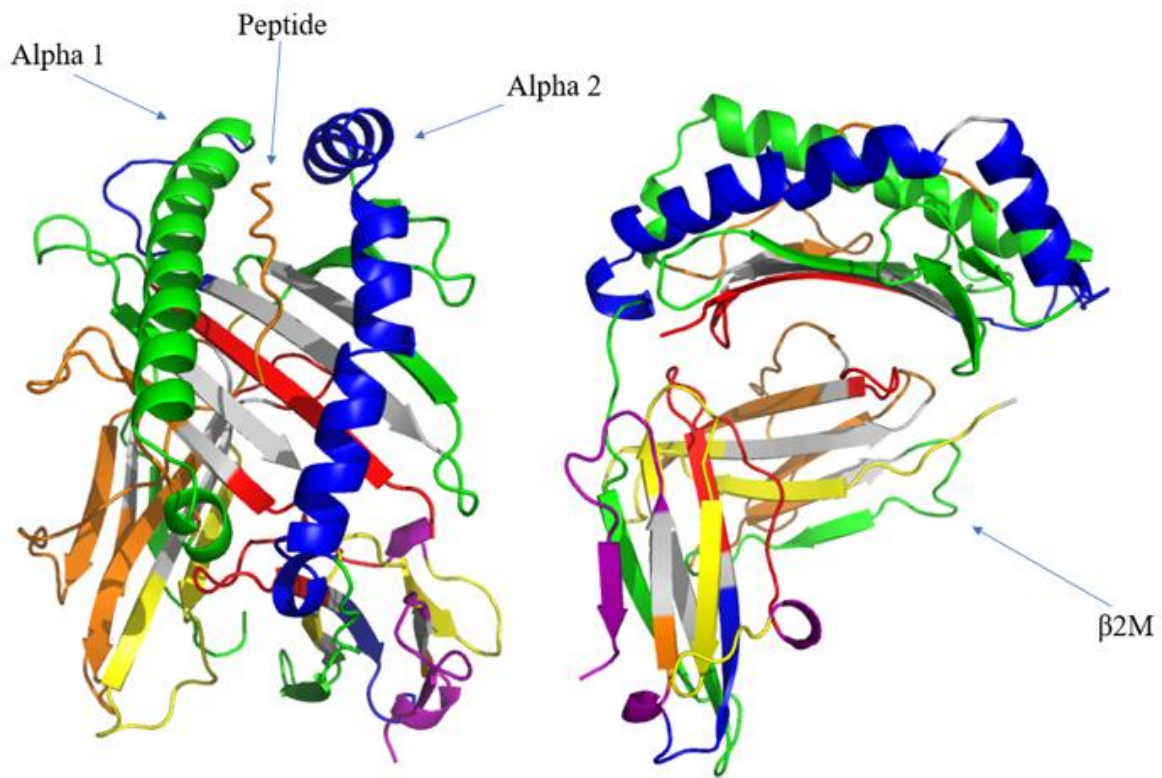
### 3.4.3 Apparent Diffusion Coefficient

This section outlines the findings of the diffusion analysis as outlined in the diffusion sections 2.2.3 and 2.2.4. All diffusion coefficients will be compared with the diffusion coefficient of bulk water using the SPC/E model of  $2.3 \text{ cm}^2/\text{s}$ . Diffusion coefficients and averages as well as hydration maps will be shown for each protein as well as comparative analysis between the three proteins.

#### 3.4.3.1 1B0G

Table 3: Diffusion Coefficients for 1B0G

Apparent Diffusion Coefficient $\text{cm}^2/\text{s}$						
Residues	Run 1	Run 2	Run 3	Average	$\sigma$	% Bulk Water
1-12	9.35E-06	1.03E-05	9.04E-06	9.57E-06	6.59E-07	42%
13-23	1.66E-05	1.78E-05	1.68E-05	1.71E-05	5.97E-07	74%
29-32	5.07E-06	6.48E-06	5.41E-06	5.65E-06	7.37E-07	25%
35-49	1.52E-05	1.63E-05	1.56E-05	1.57E-05	5.56E-07	68%
50-84 (Alpha 1)	1.73E-05	1.86E-05	1.73E-05	1.77E-05	7.32E-07	77%
85-94	1.76E-05	1.94E-05	1.77E-05	1.82E-05	9.73E-07	79%
104-113	1.72E-05	1.84E-05	1.75E-05	1.77E-05	6.52E-07	77%
118-137	1.69E-05	1.85E-05	1.70E-05	1.75E-05	9.00E-07	76%
138-179 (Alpha 2)	1.84E-05	1.96E-05	1.85E-05	1.88E-05	6.41E-07	82%
180-200	1.72E-05	1.84E-05	1.71E-05	1.76E-05	7.26E-07	76%
206-224	1.54E-05	1.70E-05	1.59E-05	1.61E-05	8.35E-07	70%
225-227 (Alpha 3)	1.86E-05	2.03E-05	1.90E-05	1.93E-05	9.03E-07	84%
228-244	1.14E-05	1.28E-05	1.15E-05	1.19E-05	7.98E-07	52%
246-253	1.74E-05	1.85E-05	1.78E-05	1.79E-05	5.34E-07	78%
254-256	1.90E-05	2.05E-05	1.93E-05	1.96E-05	8.11E-07	85%
257-258	1.51E-05	1.62E-05	1.54E-05	1.55E-05	5.57E-07	68%
262-275	1.85E-05	2.00E-05	1.87E-05	1.91E-05	8.43E-07	83%
276-298	1.58E-05	1.70E-05	1.58E-05	1.62E-05	6.81E-07	70%
307-330	1.52E-05	1.65E-05	1.55E-05	1.57E-05	6.66E-07	68%
332-337	1.20E-05	1.41E-05	1.22E-05	1.27E-05	1.15E-06	55%
339-355	1.55E-05	1.66E-05	1.57E-05	1.59E-05	6.12E-07	69%
359-375	1.66E-05	1.77E-05	1.66E-05	1.70E-05	6.66E-07	74%
376-384 (Peptide)	1.54E-05	1.62E-05	1.45E-05	1.54E-05	8.52E-07	67%



Hydration Map Color Key

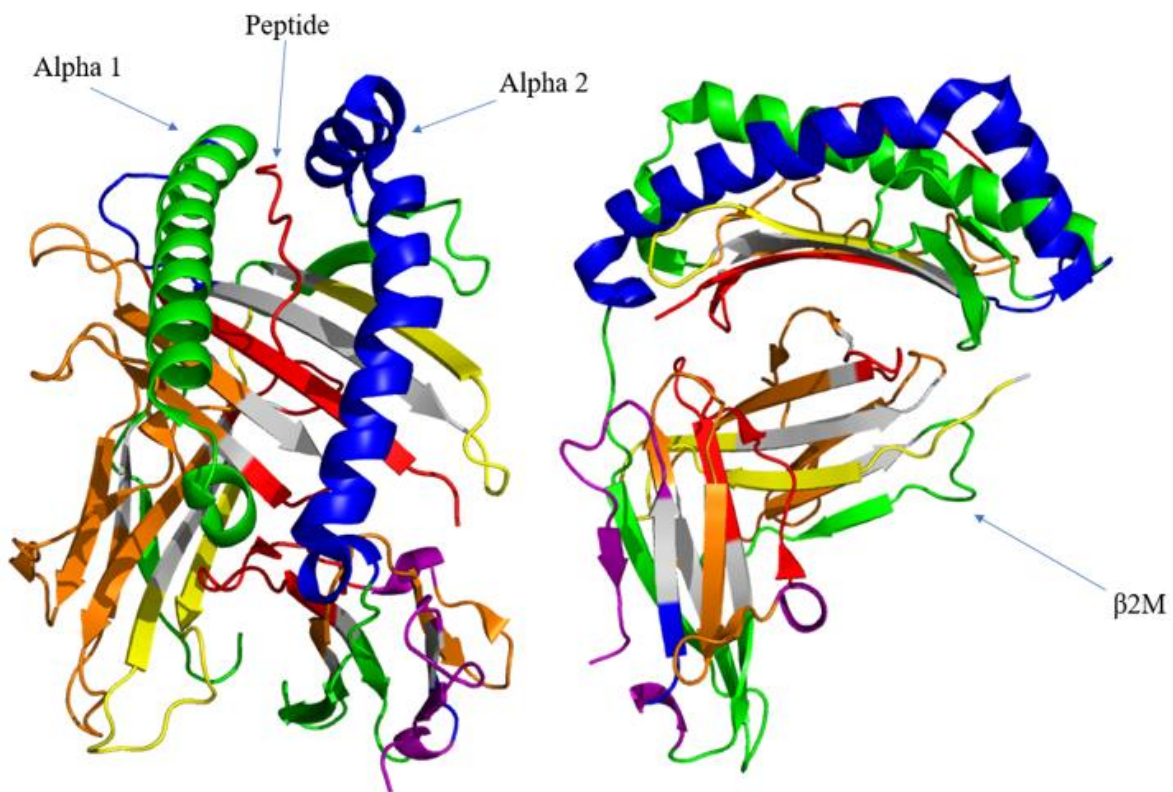
Red	$< 1.50 \times 10^{-5} \text{ cm}^2/\text{s}$
Orange	$1.5-1.59 \times 10^{-5} \text{ cm}^2/\text{s}$
Yellow	$1.60-1.69 \times 10^{-5} \text{ cm}^2/\text{s}$
Green	$1.70-1.79 \times 10^{-5} \text{ cm}^2/\text{s}$
Blue	$1.80-1.89 \times 10^{-5} \text{ cm}^2/\text{s}$
Purple	$> 1.90 \times 10^{-5} \text{ cm}^2/\text{s}$

Figure 11: Hydration Map for 1B0G

### 3.4.3.2 1EEZ

Table 4: Diffusion Coefficients for 1EEZ

Apparent Diffusion Coefficient cm <sup>2</sup> /s						
Residues	Run 1	Run 2	Run 3	Average	$\sigma$	% Bulk Water
1-12	8.02E-06	8.58E-06	1.03E-05	8.96E-06	1.17E-06	39%
13-25	1.48E-05	1.53E-05	1.64E-05	1.55E-05	8.31E-07	67%
29-32	5.38E-06	5.10E-06	7.09E-06	5.86E-06	1.08E-06	25%
34-48	1.47E-05	1.45E-05	1.63E-05	1.51E-05	1.00E-06	66%
49-85 (Alpha 1)	1.72E-05	1.69E-05	1.87E-05	1.76E-05	9.40E-07	77%
86-94	1.82E-05	1.81E-05	1.99E-05	1.88E-05	1.03E-06	82%
104-115	1.62E-05	1.64E-05	1.79E-05	1.68E-05	9.35E-07	73%
118-136	1.65E-05	1.67E-05	1.81E-05	1.71E-05	8.66E-07	74%
137-180 (Alpha 2)	1.82E-05	1.81E-05	1.95E-05	1.86E-05	7.55E-07	81%
181-200	1.71E-05	1.72E-05	1.88E-05	1.77E-05	9.28E-07	77%
206-224	1.55E-05	1.54E-05	1.71E-05	1.60E-05	9.33E-07	69%
225-228 (Alpha 3)	1.83E-05	1.84E-05	2.02E-05	1.90E-05	1.06E-06	83%
229-244	1.17E-05	1.15E-05	1.33E-05	1.22E-05	9.59E-07	53%
247-252	1.72E-05	1.74E-05	1.85E-05	1.77E-05	7.13E-07	77%
253-255	1.88E-05	1.88E-05	2.05E-05	1.94E-05	1.00E-06	84%
256-258	1.84E-05	1.86E-05	1.97E-05	1.89E-05	6.87E-07	82%
262-275	1.86E-05	1.85E-05	2.02E-05	1.91E-05	9.61E-07	83%
276-298	1.60E-05	1.60E-05	1.76E-05	1.65E-05	9.54E-07	72%
307-330	1.55E-05	1.53E-05	1.69E-05	1.59E-05	8.70E-07	69%
332-337	1.19E-05	1.21E-05	1.38E-05	1.26E-05	1.04E-06	55%
339-355	1.54E-05	1.54E-05	1.69E-05	1.59E-05	8.53E-07	69%
359-375	1.69E-05	1.69E-05	1.82E-05	1.73E-05	7.74E-07	75%
376-384 (Peptide)	1.32E-05	1.33E-05	1.45E-05	1.37E-05	7.29E-07	60%



Hydration Map Color Key

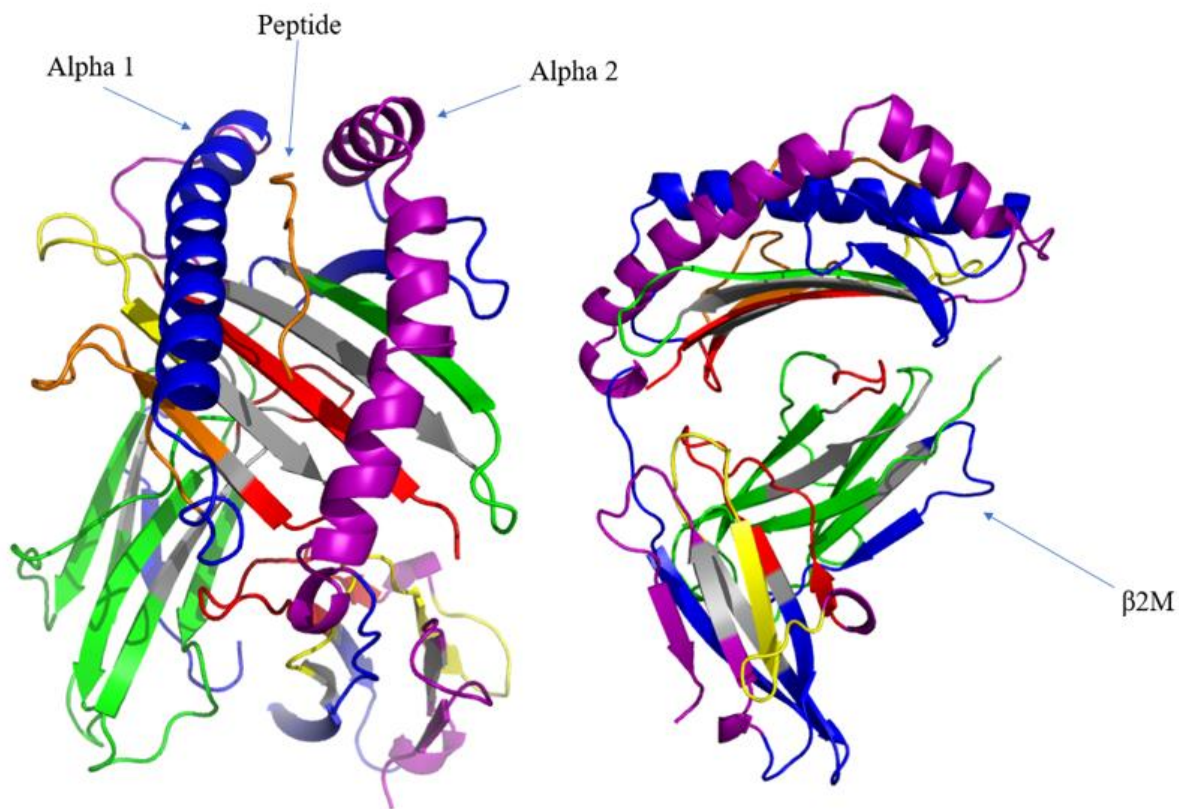
Red	$< 1.50 \times 10^{-5} \text{ cm}^2/\text{s}$
Orange	$1.5 - 1.59 \times 10^{-5} \text{ cm}^2/\text{s}$
Yellow	$1.60 - 1.69 \times 10^{-5} \text{ cm}^2/\text{s}$
Green	$1.70 - 1.79 \times 10^{-5} \text{ cm}^2/\text{s}$
Blue	$1.80 - 1.89 \times 10^{-5} \text{ cm}^2/\text{s}$
Purple	$> 1.90 \times 10^{-5} \text{ cm}^2/\text{s}$

Figure 12: Hydration Map for 1EEZ

### 3.4.3.3 111F

Table 5: Diffusion Coefficients for 111F

Apparent Diffusion Coefficient cm <sup>2</sup> /s						
Residues	Run 1	Run 2	Run 3	Average	σ	% Bulk Water
1-12	8.71E-06	1.00E-05	1.15E-05	1.01E-05	1.41E-06	43.84%
13-23	1.53E-05	1.70E-05	1.81E-05	1.68E-05	1.43E-06	72.94%
29-32	5.10E-06	5.90E-06	8.09E-06	6.36E-06	1.55E-06	27.67%
34-48	1.43E-05	1.64E-05	1.72E-05	1.60E-05	1.50E-06	69.48%
49-84 (Alpha 1)	1.68E-05	1.89E-05	1.97E-05	1.85E-05	1.49E-06	80.33%
85-94	1.78E-05	2.01E-05	2.04E-05	1.94E-05	1.40E-06	84.55%
104-116	1.59E-05	1.79E-05	1.87E-05	1.75E-05	1.43E-06	76.06%
118-137	1.62E-05	1.84E-05	1.93E-05	1.80E-05	1.57E-06	78.20%
138-179 (Alpha 2)	1.79E-05	1.99E-05	2.08E-05	1.95E-05	1.48E-06	84.90%
180-200	1.70E-05	1.89E-05	1.93E-05	1.84E-05	1.20E-06	79.95%
206-224	1.53E-05	1.71E-05	1.79E-05	1.68E-05	1.37E-06	72.90%
225-227 (Alpha 3)	1.86E-05	2.02E-05	2.11E-05	1.99E-05	1.28E-06	86.67%
228-244	1.14E-05	1.28E-05	1.41E-05	1.28E-05	1.37E-06	55.49%
246-253	1.70E-05	1.87E-05	1.96E-05	1.84E-05	1.29E-06	80.19%
254-255	1.75E-05	1.95E-05	2.04E-05	1.91E-05	1.47E-06	83.23%
256-258	1.81E-05	2.05E-05	2.10E-05	1.98E-05	1.56E-06	86.26%
262-275	1.83E-05	2.04E-05	2.11E-05	1.99E-05	1.48E-06	86.60%
276-298	1.61E-05	1.75E-05	1.87E-05	1.74E-05	1.34E-06	75.82%
307-330	1.50E-05	1.72E-05	1.79E-05	1.67E-05	1.51E-06	72.61%
332-337	1.18E-05	1.39E-05	1.51E-05	1.36E-05	1.71E-06	59.19%
339-355	1.53E-05	1.66E-05	1.80E-05	1.66E-05	1.37E-06	72.31%
359-375	1.66E-05	1.87E-05	1.93E-05	1.82E-05	1.40E-06	79.03%
376-384 (Peptide)	1.34E-05	1.58E-05	1.59E-05	1.50E-05	1.42E-06	65.27%



Hydration Map Color Key

Red	$< 1.50 \times 10^{-5} \text{ cm}^2/\text{s}$
Orange	$1.5-1.59 \times 10^{-5} \text{ cm}^2/\text{s}$
Yellow	$1.60-1.69 \times 10^{-5} \text{ cm}^2/\text{s}$
Green	$1.70-1.79 \times 10^{-5} \text{ cm}^2/\text{s}$
Blue	$1.80-1.89 \times 10^{-5} \text{ cm}^2/\text{s}$
Purple	$> 1.90 \times 10^{-5} \text{ cm}^2/\text{s}$

Figure 13: Hydration Map for 111F

#### 3.4.3.4 All Diffusion

After conducting the student's T test, it was determined that the diffusion coefficients for all regions show no significant variation at the 95% confidence interval. This consistency in the hydration layer across the three MHC proteins examined underscores the critical role of the hydration layer in molecular recognition. The hydration layer's stability and uniformity suggest a fundamental aspect of the molecular recognition process, potentially influencing the binding affinity and specificity of MHC proteins. This insight provides a deeper understanding of the biophysical properties governing antigen presentation and immune response. Furthermore, these findings open avenues for more targeted approaches in immunotherapy and vaccine design, where manipulating the hydration layer could enhance the efficacy and specificity of MHC-related immune responses. Among various residues studied, residues 256(7)-258 stood out as an anomaly, the potential cause of this is most likely due to its lack of secondary structure and number of residues analyzed due to exposure to solvent. 1B0G had statistically significant differences in diffusion in this region when compared with 1EEZ and 1I1F which both have an extra residue exposed to water. 1EEZ and 1I1F, when compared with each other show no statistically significant difference in diffusion coefficients.

### 3.4.4 RDF

#### 3.4.4.1 1B0G

The RDF plot shown in Figure 15 shows a distinct solvation shell which ends at 6 Å. The script `rdf_integrate.sh` was used to integrate all plots using GROMACS. Table 6 shows integrals from all simulations and the average of the three.

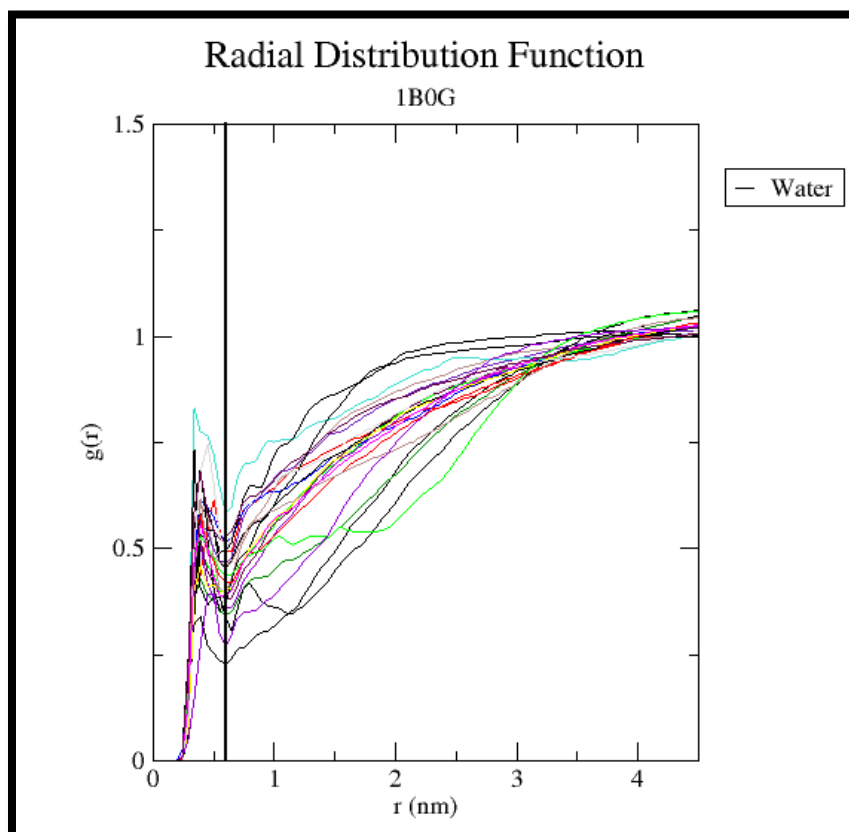


Figure 14: RDF Plot for 1B0G

Table 6: RDF Integrals for 1BOG

RDF Integral					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	9.04E-02	8.44E-02	8.57E-02	8.68E-02	3.15E-03
13-23	1.64E-01	1.60E-01	1.63E-01	1.62E-01	2.25E-03
29-32	1.32E-01	1.48E-01	1.41E-01	1.40E-01	7.87E-03
35-49	1.49E-01	1.49E-01	1.48E-01	1.48E-01	5.35E-04
50-84 (Alpha 1)	1.35E-01	1.32E-01	1.35E-01	1.34E-01	1.43E-03
85-94	1.71E-01	1.76E-01	1.72E-01	1.73E-01	2.86E-03
104-113	1.49E-01	1.58E-01	1.49E-01	1.52E-01	5.57E-03
118-137	1.30E-01	1.33E-01	1.26E-01	1.30E-01	3.34E-03
138-179 (Alpha 2)	1.29E-01	1.47E-01	1.31E-01	1.35E-01	9.86E-03
180-200	1.45E-01	1.55E-01	1.44E-01	1.48E-01	6.01E-03
206-224	1.31E-01	1.50E-01	1.33E-01	1.38E-01	1.04E-02
225-227 (Alpha 3)	2.09E-01	2.58E-01	2.18E-01	2.28E-01	2.62E-02
228-244	1.11E-01	1.21E-01	1.13E-01	1.15E-01	5.60E-03
246-253	1.28E-01	1.45E-01	1.28E-01	1.33E-01	1.00E-02
254-256	1.56E-01	1.82E-01	1.61E-01	1.66E-01	1.36E-02
257-258	7.09E-02	9.31E-02	7.37E-02	7.93E-02	1.21E-02
262-275	1.45E-01	1.65E-01	1.53E-01	1.54E-01	1.00E-02
276-298	1.44E-01	1.39E-01	1.41E-01	1.41E-01	2.35E-03
307-330	1.44E-01	1.47E-01	1.49E-01	1.47E-01	2.42E-03
332-337	1.41E-01	1.38E-01	1.40E-01	1.40E-01	1.69E-03
339-355	1.18E-01	1.22E-01	1.16E-01	1.19E-01	2.83E-03
359-375	1.74E-01	1.75E-01	1.66E-01	1.71E-01	5.01E-03
376-384 (Peptide)	8.49E-02	8.71E-02	1.00E-01	9.07E-02	8.32E-03

### 3.4.4.2 1EEZ

The RDF plot shown in Figure 16 shows a distinct solvation shell which ends at 6 Å. The script `rdf_integrate.sh` was used to integrate all plots using GROMACS. Table 7 shows integrals from all simulations and the average of the three.

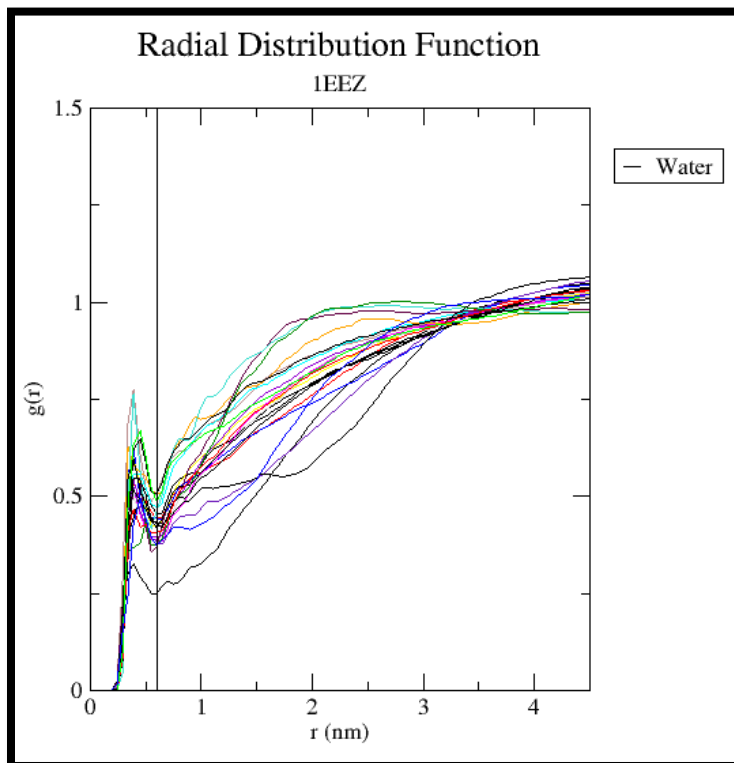


Figure 15: RDF Plot for 1EEZ

Table 7: RDF Integrals for 1EEZ

<b>RDF Integrals</b>					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	8.95E-02	9.09E-02	9.13E-02	9.06E-02	9.54E-04
13-25	1.41E-01	1.41E-01	1.40E-01	1.41E-01	4.56E-04
29-32	1.39E-01	1.38E-01	1.48E-01	1.42E-01	5.25E-03
34-48	1.48E-01	1.45E-01	1.46E-01	1.47E-01	1.44E-03
49-85 (Alpha 1)	1.46E-01	1.45E-01	1.42E-01	1.44E-01	2.21E-03
86-94	1.91E-01	1.82E-01	1.84E-01	1.85E-01	4.65E-03
104-115	1.32E-01	1.37E-01	1.40E-01	1.36E-01	3.79E-03
118-136	1.27E-01	1.26E-01	1.27E-01	1.27E-01	5.34E-04
137-180 (Alpha 2)	1.37E-01	1.32E-01	1.36E-01	1.35E-01	2.86E-03
181-200	1.51E-01	1.49E-01	1.52E-01	1.50E-01	1.57E-03
206-224	1.32E-01	1.32E-01	1.34E-01	1.33E-01	9.37E-04
225-228 (Alpha 3)	1.68E-01	1.60E-01	1.86E-01	1.71E-01	1.31E-02
229-244	1.18E-01	1.29E-01	1.29E-01	1.25E-01	6.28E-03
247-252	1.34E-01	1.32E-01	1.32E-01	1.32E-01	1.05E-03
253-255	1.49E-01	1.48E-01	1.65E-01	1.54E-01	9.35E-03
256-258	1.05E-01	1.11E-01	1.04E-01	1.07E-01	3.68E-03
262-275	1.55E-01	1.55E-01	1.55E-01	1.55E-01	3.40E-04
276-298	1.29E-01	1.36E-01	1.39E-01	1.35E-01	4.96E-03
307-330	1.53E-01	1.47E-01	1.45E-01	1.48E-01	4.17E-03
332-337	1.36E-01	1.41E-01	1.30E-01	1.36E-01	5.42E-03
339-355	1.21E-01	1.21E-01	1.21E-01	1.21E-01	2.05E-04
359-375	1.48E-01	1.62E-01	1.57E-01	1.56E-01	7.04E-03
376-384 (Peptide)	1.39E-01	1.18E-01	1.22E-01	1.26E-01	1.12E-02

### 3.4.4.3 111F

The RDF plot shown in Figure 17 shows a distinct solvation shell that ends at  $6 \text{ \AA}$ . The script `rdf_integrate.sh` was used to integrate all plots using GROMACS. Table 8 shows integrals from all simulations and the average of the three.

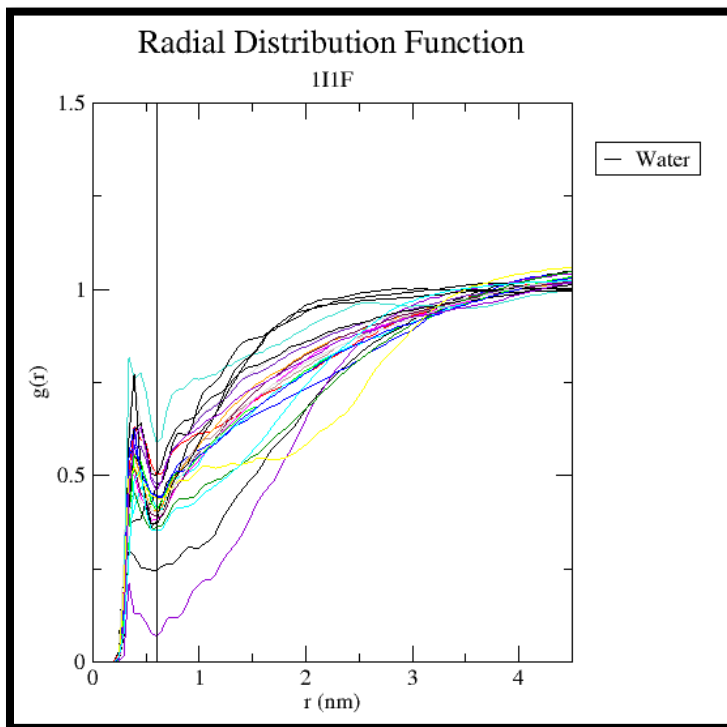


Figure 16: RDF Plot for 111F

Table 8: RDF Integrals for 1I1F

RDF Integrals					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	9.19E-02	8.68E-02	9.39E-02	9.09E-02	3.67E-03
13-23	1.60E-01	1.66E-01	1.58E-01	1.61E-01	4.17E-03
29-32	1.41E-01	1.39E-01	1.38E-01	1.39E-01	1.96E-03
34-48	1.45E-01	1.47E-01	1.43E-01	1.45E-01	2.26E-03
49-84 (Alpha 1)	1.40E-01	1.38E-01	1.41E-01	1.39E-01	1.58E-03
85-94	1.83E-01	1.81E-01	1.80E-01	1.81E-01	1.58E-03
104-116	1.34E-01	1.27E-01	1.32E-01	1.31E-01	3.80E-03
118-137	1.25E-01	1.27E-01	1.33E-01	1.29E-01	4.06E-03
138-179 (Alpha 2)	1.35E-01	1.33E-01	1.35E-01	1.34E-01	8.15E-04
180-200	1.50E-01	1.53E-01	1.51E-01	1.51E-01	1.48E-03
206-224	1.35E-01	1.34E-01	1.33E-01	1.34E-01	9.54E-04
225-227 (Alpha 3)	2.33E-01	2.10E-01	2.30E-01	2.24E-01	1.23E-02
228-244	1.14E-01	1.16E-01	1.15E-01	1.15E-01	6.86E-04
246-253	1.30E-01	1.30E-01	1.28E-01	1.29E-01	9.25E-04
254-255	1.52E-01	1.53E-01	1.46E-01	1.50E-01	3.75E-03
256-258	1.14E-01	1.11E-01	1.11E-01	1.12E-01	1.69E-03
262-275	1.57E-01	1.54E-01	1.55E-01	1.55E-01	1.53E-03
276-298	1.35E-01	1.35E-01	1.39E-01	1.36E-01	2.47E-03
307-330	1.49E-01	1.48E-01	1.47E-01	1.48E-01	8.67E-04
332-337	1.45E-01	1.41E-01	1.39E-01	1.42E-01	3.15E-03
339-355	1.19E-01	1.21E-01	1.19E-01	1.20E-01	7.69E-04
359-375	1.56E-01	1.54E-01	1.56E-01	1.55E-01	1.45E-03
376-384 (Peptide)	1.24E-01	1.16E-01	1.20E-01	1.20E-01	4.14E-03

### 3.4.4.4 RDF Comparison for Key Domains

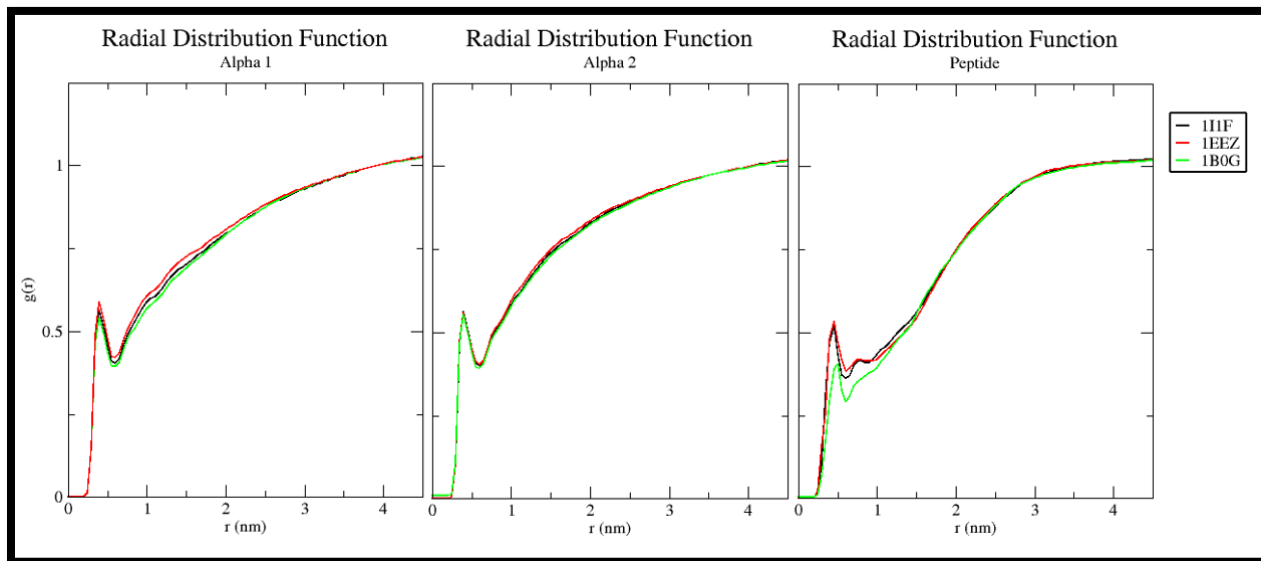


Figure 17: Comparison of RDF for Alpha1 (Left), Alpha 2 (center), and Peptide (right)

Illustrated in Figure 18 are the average Radial Distribution Function (RDF) plots for the TCR (T cell receptor) binding region, which comprises the alpha 1, alpha 2, and the peptide components of the MHC Class 1 proteins. The RDF provides a measure of the probability of finding a water molecule at a specific distance from a given reference point, thus offering key insights into the solvation environment of the protein. Upon observing these plots, it is noteworthy that the population of water molecules surrounding the two alpha helices appears to be consistent across all three proteins, suggesting a conservation of the hydration shell in these regions. The consistent hydration pattern might be indicative of the hydrophilic nature of the exposed residues in the alpha helices, allowing for the formation of a stable network of water molecules through hydrogen bonding or van der Waals interactions. In contrast, the peptide binding groove presents more variation in the water population. This could potentially be attributed to alterations in the peptide's hydrophobic nature or its ability to participate in

hydrogen bonding. Depending on the amino acid sequence of the peptide, these changes can significantly influence the hydrophilicity or hydrophobicity of the groove, and thus the solvation dynamics. In some instances, peptides with a greater proportion of hydrophobic residues might deter water from populating the groove, resulting in a less dense hydration shell. Alternatively, peptides that can actively engage in hydrogen bonding may attract and retain a more significant number of water molecules, leading to a denser hydration shell. These dynamic hydration patterns in the peptide binding groove could potentially play a role in peptide recognition and binding, and thereby influence the overall function of the MHC Class 1 proteins. In addition to the TCR binding region, we observe considerable variability in the water population surrounding the alpha 3 domain of the MHC Class 1 proteins as shown in Figure 19.

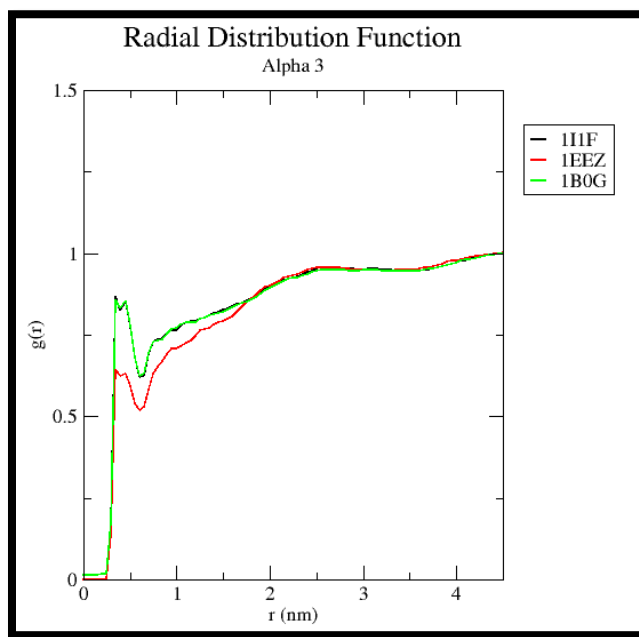


Figure 18: RDF Comparison of Alpha 3 domain

This domain is of great interest, as it serves as the binding site for the co-receptor CD8, a crucial player in the T cell response. As such, the dynamics of water in this region can potentially impact

the binding and interaction of CD8, thereby influencing the overall immune response. Fluctuations in water population around the alpha 3 domain may be attributed to several factors. For instance, it could be driven by the changes in the conformational state of the domain, alterations in the domain's electrostatic properties, or shifts in the local distribution of hydrophobic and hydrophilic residues. These factors, individually or collectively, could significantly influence the domain's solvation dynamics, leading to observable variations in the water population. Notably, changes in water population could directly or indirectly modulate the binding of CD8. In a direct context, the hydration state of the alpha 3 domain could potentially influence the binding affinity and kinetics of CD8. Indirectly, alterations in the hydration shell may lead to changes in the structural flexibility and dynamics of the domain, which in turn could impact the binding and interaction of CD8. Given the crucial role of the CD8 co-receptor in T cell activation and the immune response, understanding the solvation dynamics around the alpha 3 domain provides valuable insights into the molecular interactions at play in these biological processes. This highlights the importance of investigating not just the protein structure and dynamics, but also the role of the surrounding solvent in shaping these interactions. Such a holistic approach can aid in building a more comprehensive picture of the protein's function in the context of its biological environment. Through these RDF plots, we gain valuable insights into the behavior of water around specific regions of the proteins, aiding us in understanding the role of water in protein conformational dynamics and function. This analysis underscores the importance of exploring solvation dynamics in the context of protein simulations, contributing towards a more holistic understanding of protein behavior and interactions in a biologically relevant environment.

### 3.4.5 Hydrogen Bond Lifetime

The lifetime of hydrogen bonds between water molecules and specific protein residues offers key insights into the nature of protein-solvent interactions, and this becomes especially fascinating when analyzing a set of three MHC Class 1 proteins. The hydrogen bond lifetimes may influence the stability and function of these proteins, shedding light on how water dynamics might vary across different proteins or different regions within the same protein. The hydrogen bond dynamics could highlight specific residues or regions in the MHC Class 1 proteins that are essential for their interaction with peptides or immune cells. For example, longer bond lifetimes could indicate regions of high stability or rigidity, crucial for maintaining the structural integrity of the peptide-binding groove or other functional sites. On the other hand, regions with shorter bond lifetimes may reveal zones of higher flexibility, which could be associated with regions participating in dynamic interactions, such as the co-receptor binding sites. These insights can be crucial for understanding the intricate biophysical processes governing these proteins' behavior and their roles in the immune response.

### 3.4.5.1 1B0G

Table 9: Hydrogen Bond Lifetime 1B0G

Hydrogen Bond Lifetime (ps)					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	27.86	30.79	34.17	30.94	3.15
13-23	23.52	22.36	20.63	22.17	1.45
29-32	50.41	51.73	43.47	48.54	4.44
35-49	33.18	30.96	28.58	30.90	2.30
50-84 (Alpha 1)	26.53	27.27	28.49	27.43	0.99
85-94	25.81	20.44	24.54	23.60	2.80
104-113	17.71	17.97	17.51	17.73	0.23
118-137	29.74	16.75	30.83	25.77	7.83
138-179 (Alpha 2)	16.14	27.65	22.01	21.94	5.75
180-200	27.31	27.79	26.78	27.29	0.51
206-224	30.88	30.99	31.46	31.11	0.31
225-227 (Alpha 3)	26.66	14.07	23.58	21.44	6.56
228-244	35.89	33.08	18.14	29.04	9.54
246-253	31.20	22.04	22.33	25.19	5.21
254-256	35.61	31.91	32.48	33.33	1.99
257-258	25.49	48.45	29.48	34.48	12.27
262-275	23.90	25.35	24.45	24.56	0.73
276-298	24.52	23.83	20.91	23.09	1.91
307-330	31.23	30.78	17.21	26.40	7.97
332-337	23.97	23.24	25.36	24.19	1.07
339-355	30.52	31.52	31.58	31.21	0.59
359-375	15.89	21.68	23.33	20.30	3.91
376-384 (Peptide)	9.31	22.71	10.10	14.04	7.52

### 3.4.5.2 1EEZ

Table 10: Hydrogen Bond Lifetime 1EEZ

<b>Hydrogen Bond Lifetime (ps)</b>					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	26.29	25.51	25.36	25.72	0.50
13-25	26.38	23.18	24.88	24.81	1.60
29-32	46.50	55.32	50.99	50.94	4.41
34-48	32.14	31.54	28.97	30.88	1.69
49-85 (Alpha 1)	28.44	29.33	29.07	28.94	0.46
86-94	24.09	23.44	22.74	23.42	0.67
104-115	21.08	20.98	18.69	20.25	1.35
118-136	28.95	28.77	27.38	28.37	0.86
137-180 (Alpha 2)	27.75	27.54	28.11	27.80	0.29
181-200	25.37	27.10	25.49	25.99	0.97
206-224	31.95	31.82	30.24	31.34	0.95
225-228 (Alpha 3)	26.97	26.56	19.89	24.47	3.97
229-244	34.18	33.57	31.85	33.20	1.20
247-252	32.91	60.36	35.50	42.92	15.16
253-255	43.05	43.45	38.96	41.82	2.48
256-258	22.01	18.55	21.33	20.63	1.83
262-275	22.27	23.78	24.60	23.55	1.18
276-298	22.82	22.18	23.29	22.76	0.56
307-330	29.65	29.42	29.78	29.62	0.18
332-337	21.78	27.09	25.85	24.90	2.78
339-355	29.33	28.92	29.23	29.16	0.21
359-375	20.68	20.75	20.93	20.79	0.13
376-384 (Peptide)	50.65	26.91	25.01	34.19	14.29

### 3.4.5.3 111F

Table 11: Hydrogen Bond Lifetime 111F

<b>Hydrogen Bond Lifetime</b>					
Residues	Run 1	Run 2	Run 3	Average	$\sigma$
1-12	28.65	28.14	24.22	27.00	2.42
13-23	29.82	23.59	28.97	27.46	3.38
29-32	47.69	49.31	48.21	48.40	0.83
34-48	30.88	31.64	32.31	31.61	0.72
49-84 (Alpha 1)	27.40	26.93	26.82	27.05	0.31
85-94	20.90	22.17	22.08	21.72	0.71
104-116	18.80	20.41	19.09	19.43	0.85
118-137	31.04	29.11	29.40	29.85	1.04
138-179 (Alpha 2)	28.74	27.51	27.13	27.79	0.84
180-200	26.33	26.84	27.41	26.86	0.54
206-224	30.61	29.67	30.21	30.16	0.47
225-227 (Alpha 3)	22.18	27.66	21.26	23.70	3.46
228-244	33.01	33.79	33.42	33.40	0.39
246-253	25.18	21.55	28.85	25.19	3.65
254-255	44.82	65.91	73.21	61.32	14.75
256-258	20.40	20.49	19.50	20.13	0.55
262-275	22.81	23.69	23.31	23.27	0.44
276-298	21.81	22.30	22.02	22.04	0.25
307-330	29.38	30.77	30.74	30.30	0.80
332-337	26.16	25.51	24.54	25.40	0.82
339-355	28.83	29.00	28.32	28.72	0.36
359-375	21.03	21.28	21.06	21.12	0.14
376-384 (Peptide)	33.21	40.78	34.44	36.15	4.06

#### 3.4.5.4 Hydrogen Bond Lifetime Summary

Like the findings regarding diffusion coefficients, hydrogen bond lifetime emerges as another notable and consistent feature shared among MHC proteins. It's worth noting that, with the exception of the loop region spanning from residues 246 to 256, the bound peptide exhibits substantial variability among these proteins, primarily attributable to the hydrogen bonding potential inherent to the peptide itself. To delve deeper into this aspect, the Structural Analysis Suite was employed to quantify the percentage of the peptide that comprises hydrophobic residues and the percentage of residues within the peptide with the capacity to form hydrogen bonds. It's apparent that the composition of the peptide sequence directly influences the hydrogen bond lifetime within the MHC proteins. For instance, the 1B0G protein consists of a notable 88% of hydrophobic residues, of which 11% have the ability to form hydrogen bonds. This high proportion of hydrophobic residues and limited hydrogen bonding capacity correlates with the protein's relatively shorter hydrogen bond lifetime. In contrast, examining the bound peptide in 1I1F reveals a different scenario, with approximately 56% of its residues being hydrophobic, and a significant 33% of them possessing the potential to form hydrogen bonds. There is a poor correlation between hydrogen bond lifetimes and diffusion coefficients; this is likely due to only sampling a fraction of water in the hydration layer. These findings emphasize the intricate relationship between peptide composition and hydrogen bond lifetime, highlighting how variations in the peptide sequence contribute to the dynamic behavior of MHC proteins. Further exploration of these relationships can provide valuable insights into the functional roles of MHC proteins in antigen presentation and immune response regulation.

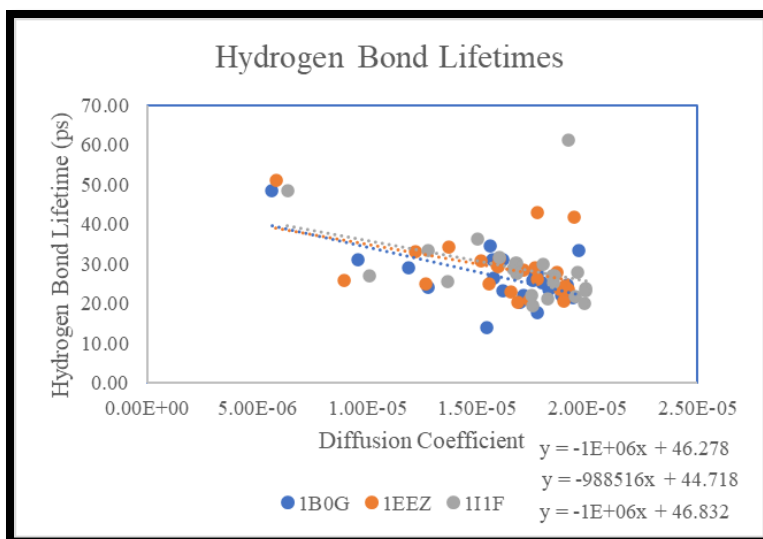


Figure 19: Comparison of Hydrogen Bond Lifetimes

Table 7: Comparison of Hydrogen Bond Lifetimes

Hydrogen Bond Lifetime (ps)					
Residues	1B0G	1EEZ	1I1F	Average	$\sigma$
1-12	30.94	25.72	27.00	27.89	2.72
13-23	22.17	24.81	27.46	24.81	2.64
29-32	48.54	50.94	48.40	49.29	1.42
35-49	30.90	30.88	31.61	31.13	0.41
50-84 (Alpha 1)	27.43	28.94	27.05	27.81	1.00
85-94	23.60	23.42	21.72	22.91	1.04
104-113	17.73	20.25	19.43	19.14	1.28
118-137	25.77	28.37	29.85	28.00	2.06
138-179 (Alpha 2)	21.94	27.80	27.79	25.84	3.38
180-200	27.29	25.99	26.86	26.71	0.67
206-224	31.11	31.34	30.16	30.87	0.62
225-227 (Alpha 3)	21.44	24.47	23.70	23.20	1.58
228-244	29.04	33.20	33.40	31.88	2.46
246-253	25.19	42.92	25.19	31.10	10.24
254-256	33.33	41.82	61.32	45.49	14.35
257-258	34.48	20.63	20.13	25.08	8.14
262-275	24.56	23.55	23.27	23.79	0.68
276-298	23.09	22.76	22.04	22.63	0.53
307-330	26.40	29.62	30.30	28.77	2.08
332-337	24.19	24.90	25.40	24.83	0.61
339-355	31.21	29.16	28.72	29.69	1.33
359-375	20.30	20.79	21.12	20.74	0.42
376-384 (Peptide)	14.04	34.19	36.15	28.12	12.24

### 3.4.6 Reorientation Times

Analyzing the reorientation times of water molecules in the vicinity of specific protein residues in a set of three MHC Class 1 proteins can provide crucial insights into the dynamic solvation environment surrounding these proteins. The reorientation time describes how quickly water molecules can change their orientation, which reflects the dynamism of the local hydration shell. This can offer valuable information about how the local protein environment might affect water behavior and thus influence protein stability, function, and interaction dynamics. Longer reorientation times might indicate areas of the protein where water interactions are more structured or ordered, possibly due to strong interactions with protein residues or other topographical aspects (concave/convex). These could potentially represent regions essential for maintaining protein stability or function. Conversely, regions characterized by shorter water reorientation times might reflect a more dynamic hydration environment, possibly corresponding to protein areas that undergo significant conformational changes or are involved in transient interactions such as ligand binding and release. This can help map out regions of interest in the MHC Class 1 proteins and provide a more nuanced understanding of how protein-water interactions influence protein behavior in the broader context of immune response. Reorientation times are given in two terms:  $\tau_1$  referring to fast “jump” reorientations and  $\tau_2$  referring to slower reorientation. The values presented align with times found by Hynes et. al., in their study of hydration shells across 6 proteins [14]. A graph plotting reorientation times vs diffusion coefficient is included for each protein both as an internal check and proof of concept that reorientation dynamics and diffusion measure the movement of water in two distinct ways and correlate well to one another.

### 3.4.6.1 1B0G

Table 8: Reorientation Times 1B0G

Reorientation Times								
Residues	Run 1		Run 2		Run 3		Average	
	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$
1-12	4.69	30.35	4.75	31.54	4.71	31.24	4.72	31.04
13-23	4.50	22.89	4.51	23.83	4.59	24.45	4.53	23.72
29-32	4.77	35.50	4.75	34.90	4.75	34.88	4.75	35.09
35-49	4.61	25.66	4.63	26.11	4.62	25.76	4.62	25.84
50-84 (Alpha 1)	4.44	21.25	4.46	21.20	4.49	21.91	4.46	21.45
85-94	4.46	21.33	7.81	7.81	4.47	21.63	5.58	16.92
104-113	4.43	21.26	4.48	21.90	8.12	8.12	5.68	17.09
118-137	4.52	22.35	4.49	22.50	4.49	22.42	4.50	22.43
138-179 (Alpha 2)	4.37	19.55	4.42	20.40	4.41	20.11	4.40	20.02
180-200	4.48	21.79	4.49	21.64	4.42	21.22	4.46	21.55
206-224	4.60	25.20	4.53	23.96	4.51	23.89	4.55	24.35
225-227 (Alpha 3)	4.41	20.19	4.38	20.11	4.36	19.33	4.38	19.88
228-244	4.76	29.03	4.69	28.44	4.70	28.13	4.71	28.53
246-253	4.54	23.53	4.56	24.40	4.48	22.54	4.53	23.49
254-256	4.45	21.16	4.48	20.91	4.35	20.33	4.43	20.80
257-258	4.55	24.43	4.64	25.22	4.55	24.26	4.58	24.64
262-275	4.34	19.02	4.25	17.59	4.28	18.31	4.29	18.31
276-298	4.55	23.50	4.55	23.77	4.51	23.54	4.53	23.60
307-330	4.56	25.20	4.54	25.00	4.57	25.07	4.56	25.09
332-337	4.62	26.15	4.65	25.84	4.60	26.39	4.62	26.13
339-355	4.52	24.87	4.54	25.37	4.53	24.50	4.53	24.91
359-375	4.54	23.37	4.53	23.00	4.50	23.12	4.52	23.16
376-384 (Peptide)	4.38	20.35	4.42	22.09	4.41	21.73	4.40	21.39

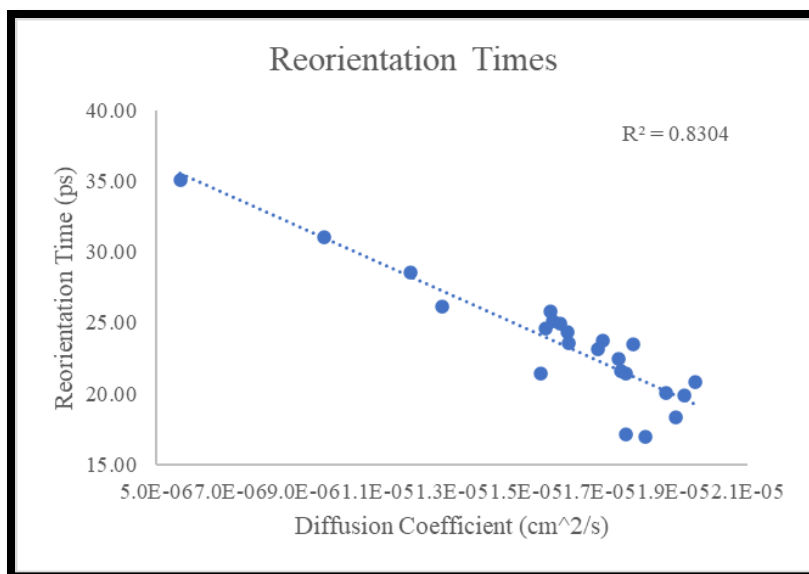


Figure 20: Reorientation Diffusion Correlation

### 3.4.6.2 1EEZ

Table 9: Reorientation Times 1EEZ

Residues	Reorientation Times							
	Run 1		Run 2		Run 3		Average	
	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$
1-12	4.76	33.34	4.74	33.03	4.74	33.28	4.75	33.22
13-25	4.55	25.77	4.55	25.61	4.63	26.85	4.58	26.08
29-32	4.84	35.61	4.88	37.21	4.97	35.59	4.90	36.14
34-48	4.62	25.76	4.60	26.18	4.62	26.06	4.61	26.00
49-85 (Alpha 1)	4.43	21.24	4.48	21.95	4.46	21.31	4.46	21.50
86-94	4.37	19.80	4.43	20.87	4.40	20.05	4.40	20.24
104-115	4.53	23.24	4.48	22.66	4.49	22.39	4.50	22.76
118-136	4.49	22.55	4.47	22.52	4.48	22.76	4.48	22.61
137-180 (Alpha 2)	7.65	7.65	7.61	7.61	4.40	20.23	6.55	11.83
181-200	4.39	20.66	4.43	20.95	4.42	20.73	4.42	20.78
206-224	4.56	23.88	4.57	24.32	4.59	24.39	4.58	24.20
225-228 (Alpha 3)	4.34	19.07	4.36	19.27	4.38	19.17	4.36	19.17
229-244	4.65	26.79	4.76	28.61	4.72	27.69	4.71	27.70
247-252	4.51	22.68	4.52	23.26	4.52	22.39	4.52	22.77
253-255	4.47	20.95	7.53	7.53	4.45	20.59	5.48	16.36
256-258	7.71	7.71	7.71	7.71	7.82	7.82	7.75	7.75
262-275	4.27	17.84	4.28	18.18	4.28	18.04	4.28	18.02
276-298	4.47	21.73	4.49	22.55	4.50	22.46	4.49	22.25
307-330	4.56	24.84	4.56	25.33	4.57	25.53	4.56	25.23
332-337	4.68	26.23	4.66	26.73	4.63	25.51	4.65	26.16
339-355	4.54	24.49	4.54	25.05	4.58	25.37	4.55	24.97
359-375	4.40	20.37	4.47	22.07	4.49	22.13	4.45	21.52
376-384 (Peptide)	4.63	27.41	4.62	26.75	4.63	27.41	4.63	27.19

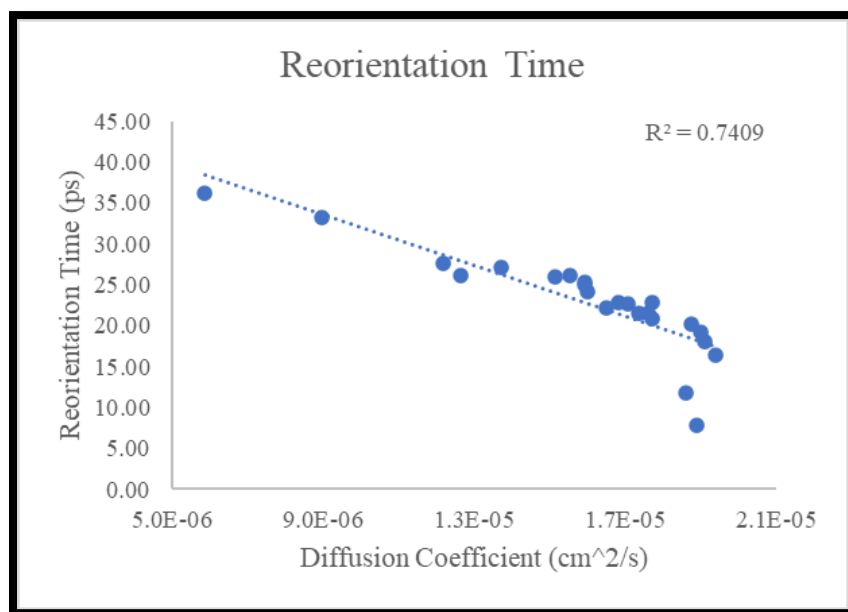


Figure 21: Reorientation Diffusion Correlation

### 3.4.6.3 1I1F

Table 15: 1I1F Reorientation Times

Reorientation Times (ps)								
Residues	Run 1		Run 2		Run 3		Average	
	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$	$\tau_1$	$\tau_2$
1-12	4.70	32.30	4.81	33.09	4.68	32.41	4.73	32.60
13-23	4.62	26.14	4.59	25.80	4.58	25.85	4.60	25.93
29-32	4.85	36.02	4.69	35.02	4.68	34.64	4.74	35.22
34-48	4.62	26.68	4.58	25.49	4.61	26.20	4.60	26.12
49-84 (Alpha 1)	4.44	21.18	4.45	21.28	4.41	20.97	4.43	21.14
85-94	4.44	21.01	4.47	21.36	4.45	21.10	4.45	21.16
104-116	4.48	23.02	4.48	22.49	4.46	22.97	4.47	22.83
118-137	4.50	23.27	4.46	22.53	4.52	22.76	4.50	22.85
138-179 (Alpha 2)	7.64	7.64	4.35	19.75	7.58	7.58	6.52	11.66
180-200	4.42	20.98	4.46	21.70	4.48	22.12	4.45	21.60
206-224	4.53	24.27	4.57	24.29	4.58	24.83	4.56	24.47
225-227 (Alpha 3)	4.36	19.81	4.44	20.24	4.46	20.52	4.42	20.19
228-244	4.68	28.02	4.69	28.22	4.67	27.98	4.68	28.07
246-253	4.50	23.43	4.49	23.75	4.48	23.16	4.49	23.45
254-255	4.53	22.83	4.43	21.97	7.88	7.88	5.61	17.56
256-258	4.48	21.72	4.36	20.20	4.38	19.99	4.41	20.63
262-275	4.27	18.39	4.35	18.70	4.26	18.01	4.29	18.37
276-298	4.46	22.02	4.53	23.16	4.53	22.77	4.51	22.65
307-330	4.60	26.10	4.55	24.98	4.56	25.27	4.57	25.45
332-337	4.64	27.14	4.65	25.81	4.59	25.34	4.63	26.09
339-355	4.53	25.03	4.57	25.14	4.57	25.56	4.55	25.24
359-375	4.49	22.55	4.46	22.15	4.51	22.65	4.49	22.45
376-384 (Peptide)	4.50	23.18	4.55	25.07	4.35	22.56	4.47	23.60

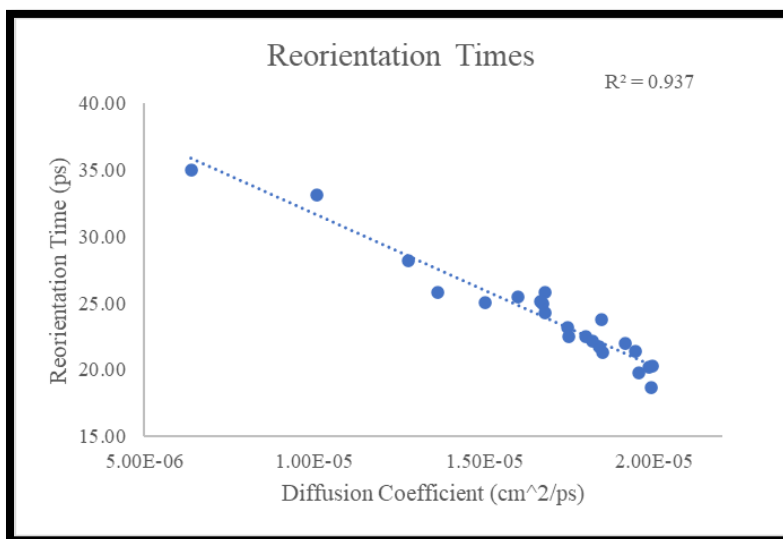


Figure 22: Reorientation Diffusion Correlation

### 3.4.7 MHC Summary

The results obtained from this research underscore the pivotal role played by the hydration shell in the context of molecular recognition. While it is well-established that the sequence of a protein governs its ultimate structure and function, it becomes increasingly evident that the hydration shell and its associated properties represent a critical factor that cannot be underestimated. The striking conservation of various hydration-related parameters, including diffusion rates, reorientation dynamics, water density, and hydrogen bond lifetime, across the entire protein, except for the specific peptide region, signifies the existence of a remarkable "sweet spot" in hydration dynamics. This sweet spot appears to be instrumental in controlling the entropic costs associated with molecular binding events, thus influencing the overall efficiency of molecular recognition processes. As we move forward, the prospect of conducting future studies with more extensive sampling of MHC proteins' hydration dynamics holds great promise. Such endeavors would undoubtedly offer valuable insights into the underlying mechanisms governing the observed trends. A broader exploration of hydration dynamics in a diverse set of MHC proteins may unveil additional nuances and refine our understanding of the interplay between protein structure, hydration, and molecular recognition. This avenue of research stands poised to contribute significantly to our comprehension of critical biological processes and could potentially inform the development of novel therapeutic strategies and interventions.

## CHAPTER 4

### KRAS DIFFUSION COEFFICIENT ANALYSIS

#### 4.1 KRAS Background

KRAS, a member of the RAS gene family, plays a critical role in cell signaling pathways, influencing cell division, growth, and death. Structurally, KRAS is a small GTPase, acting like a molecular switch. In its active state, bound to GTP (guanosine triphosphate), KRAS transmits signals from outside the cell to the cell's nucleus, promoting growth and division. When it binds to GDP (guanosine diphosphate), it becomes inactive, stopping these signals. This regulation is crucial for normal cellular functions. However, mutations in the KRAS gene can disrupt this balance. These mutations often lead to the KRAS protein being permanently active, continuously sending growth signals irrespective of external cues. This unregulated cell proliferation is a hallmark of cancer. KRAS mutations are particularly prevalent in several cancer types, including lung, colorectal, and pancreatic cancers, making it a significant focus in oncology research for targeted therapies. These mutations, usually occurring at specific hotspots like codons 12, 13, and 61, make KRAS one of the most mutated oncogenes in human cancers [15].

#### 4.2 Mutations and Simulation Details

Before initiating a molecular dynamics (MD) simulation, the acquisition of a complete structural file is imperative, which is typically sourced from RCSB.org. This task presented considerable challenges in the case of KRAS, as most available structures were flawed due to either having unmodeled segments of the protein, mutations, or a combination thereof. The intricate process of accurately modeling single residues becomes exponentially complex, particularly when the unmodeled segments extend over several residues and are not situated at

the protein's termini. After extensive deliberation, the structure identified as PDB ID 6ASE was selected as the starting point. Despite its mutation A69G, it was deemed suitable because it was fully modeled. The investigation into the diffusion properties of KRAS necessitated several mutations, not only to derive a wild-type (WT) starting structure but also to replicate various mutations as performed in the research conducted by Franck and coworkers [16]. The implementation of point mutations was adeptly facilitated using PyMOL and its mutagenesis wizard tool. A series of unique starting structures were meticulously crafted for the simulations, encompassing WT, WT with the I36T mutation, WT with the M67T mutation, 6ASE with the A59G mutation, 6ASE with dual mutations A59G and I35T, and 6ASE with the mutations A59G and M67T, culminating in six distinct starting configurations. To ensure statistical robustness, each protein variant underwent triplicate simulation runs, the results of which were averaged to draw comprehensive conclusions. The simulations conducted for this study were carried out using the GROMACS software in conjunction with the amber03 forcefield and the SPC/E water model. The equilibration phase was executed in three steps, comprising an initial energy minimization step followed by a 1000 picosecond NVT equilibration phase to stabilize the temperature and a subsequent 5,000 picosecond NPT equilibration phase to achieve the desired pressure conditions. Throughout these equilibration phases, position restraints were thoughtfully applied to maintain system integrity. Following the equilibration phase, a 100 nanosecond (ns) production run was initiated under the NVT ensemble to collect valuable data and insights. This approach ensures a constant number of particles, volume, and temperature during the simulation, mimicking realistic conditions for the system under investigation.

### 4.3 Residue Group Determination

For the analysis conducted in this study, we harnessed the powerful capabilities of the Fast Analysis Suite, a versatile toolkit designed to facilitate comprehensive investigations in structural biology. To assess the solvent-accessible regions within our target systems, we leveraged the user-friendly interface provided by the website <https://curie.utmb.edu/getarea.html>, which served as an indispensable resource for computing solvent-accessible surface area (SASA). Our primary focus during this phase of the analysis was on SASA measurements, as they allowed us to gain initial insights into how mutations might influence the global hydration shell surrounding the biomolecular structures under investigation. Furthermore, beyond the broad examination of SASA, we recognized the importance of delving deeper into specific secondary structure analyses. This additional layer of investigation holds significant promise, as previous research has unveiled intriguing phenomena concerning the dynamics of hydration shells in response to mutated residues within specific secondary structural elements. Understanding these dynamics at an atomistic level can provide valuable clues about the intricate interplay between molecular structure and hydration, shedding light on the broader implications of mutations in the context of biomolecular function. In order to conduct these analyses efficiently, all the necessary scripts were meticulously generated using the Bash Script Generator application, ensuring reproducibility and accuracy throughout the study. This approach allows a streamline analytical process and helps maintain control over the computational procedures, thereby enhancing the reliability and reproducibility of findings.

## 4.4 Apparent Diffusion Coefficient

### 4.4.1 Wild Type KRAS

The investigation into the hydration dynamics of KRAS has unveiled intriguing observations regarding its aqueous environment. Notably, the hydration layer of KRAS exhibits an overall shift towards the more rapid diffusion behavior of bulk water, which sets it apart from the majority of proteins. Water molecules in the first solvation shell of KRAS are shifted towards that of bulk water, ranging from 58% to 89% of its diffusion. To place this discovery into a broader context, it is illuminating to contrast these findings with those presented in the previous chapter, which delved into the hydration properties of MHC proteins. In stark contrast to the relatively narrow range of hydration levels observed in KRAS, MHC proteins exhibited a broader spectrum of hydration, ranging from approximately 25% to 84% of bulk water density. These findings find support in the pioneering work of Franck, et al., whose electron spin resonance spectroscopy study [17] provided valuable confirmation of the hydration dynamics observed in KRAS. Franck's research further contributes to our evolving comprehension of the intricate relationship between protein hydration and function, shedding light on the diverse ways in which proteins interact with their surroundings. This intriguing disparity underscores the unique hydration characteristics inherent to different proteins and highlights the need for a comprehensive understanding of their aqueous environments

Table 10: Diffusion Coefficients for Wild Type KRAS.

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	WT Run 1	WT Run 2	WT Run 3	Average	SD
1-9	1.78E-05	1.73E-05	1.84E-05	1.78E-05	5.36E-07
10-14	1.31E-05	1.41E-05	1.31E-05	1.34E-05	6.20E-07
15-26	1.71E-05	1.46E-05	1.78E-05	1.65E-05	1.72E-06
27-37	1.93E-05	1.97E-05	1.97E-05	1.95E-05	2.54E-07
38-50	1.96E-05	1.92E-05	2.07E-05	1.98E-05	7.91E-07
56-63	1.58E-05	1.60E-05	1.58E-05	1.58E-05	1.17E-07
64-75	1.95E-05	1.86E-05	2.09E-05	1.97E-05	1.15E-06
86-105	1.91E-05	1.88E-05	1.98E-05	1.92E-05	5.55E-07
106-110	1.74E-05	1.67E-05	1.86E-05	1.76E-05	9.94E-07
116-125	1.99E-05	1.92E-05	2.05E-05	1.99E-05	6.31E-07
126-138	2.02E-05	1.99E-05	2.12E-05	2.05E-05	6.85E-07
139-143	1.50E-05	1.46E-05	1.55E-05	1.50E-05	4.54E-07
146-150	1.89E-05	1.75E-05	2.02E-05	1.89E-05	1.35E-06
151-169	1.86E-05	1.76E-05	1.93E-05	1.85E-05	8.14E-07

#### 4.4.2 I36T and M67T Mutants

The examination of mutations at both residues has provided valuable insights into the hydration dynamics of the entire protein. These mutations have consistently yielded observations indicating an increased level of hydration dynamics that extends throughout the entire protein structure. To gain a more granular understanding of these effects, we conducted isolated diffusion analyses focused specifically on residues 36 and 67, which revealed results that aligned with the broader trends observed across the protein. However, it's worth noting an interesting discrepancy between our computational findings and the dynamic analysis conducted by Franck. Franck's work revealed that mutating residue 36 led to an increase in diffusion specifically around this residue, while the mutation at residue 67 appeared to induce a deceleration in the movement of water molecules around residue 67 [17]. This intriguing contrast could potentially be attributed to a critical difference between our computational simulations and Franck's experimental setup. It is essential to acknowledge that our simulations were conducted on a non-GDP-bound KRAS structure, while Franck's experiments involved a GDP-bound structure. This

distinction may account for the divergent observations and emphasizes the complex interplay between the specific structural state of KRAS and its dynamic interactions with the surrounding solvent.

Table 11: Diffusion Coefficients for I36T Mutant

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	I36T Run 1	I36T Run 2	I36T Run 3	Average	SD
1-9	1.72E-05	1.80E-05	1.90E-05	1.80E-05	8.78E-07
10-14	1.49E-05	1.29E-05	1.42E-05	1.40E-05	1.01E-06
15-26	1.57E-05	1.73E-05	1.74E-05	1.68E-05	9.56E-07
27-37	2.08E-05	1.95E-05	2.11E-05	2.05E-05	8.57E-07
38-50	1.91E-05	2.04E-05	2.08E-05	2.01E-05	8.97E-07
56-63	1.69E-05	1.60E-05	1.70E-05	1.66E-05	5.82E-07
64-75	1.94E-05	2.07E-05	2.14E-05	2.05E-05	9.93E-07
86-105	1.93E-05	1.95E-05	2.04E-05	1.97E-05	5.75E-07
106-110	1.77E-05	1.84E-05	1.89E-05	1.83E-05	6.10E-07
116-125	2.08E-05	2.03E-05	2.14E-05	2.09E-05	5.32E-07
126-138	2.06E-05	2.10E-05	2.18E-05	2.11E-05	5.93E-07
139-143	1.57E-05	1.60E-05	1.70E-05	1.62E-05	6.66E-07
146-150	1.94E-05	1.99E-05	2.05E-05	1.99E-05	5.15E-07
151-169	1.89E-05	1.94E-05	1.99E-05	1.94E-05	5.25E-07

Table 12: Diffusion Coefficients for M67T Mutant

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	M67T Run 1	M67T Run 2	M67T Run 3	Average	SD
1-9	1.81E-05	1.90E-05	1.84E-05	1.85E-05	4.55E-07
10-14	1.42E-05	1.53E-05	1.44E-05	1.46E-05	5.95E-07
15-26	1.71E-05	1.88E-05	1.72E-05	1.77E-05	9.50E-07
27-37	2.12E-05	2.07E-05	1.98E-05	2.06E-05	6.75E-07
38-50	2.04E-05	2.09E-05	2.03E-05	2.06E-05	3.39E-07
56-63	1.64E-05	1.73E-05	1.63E-05	1.66E-05	5.83E-07
64-75	2.10E-05	2.10E-05	2.04E-05	2.08E-05	3.34E-07
86-105	1.98E-05	2.03E-05	1.95E-05	1.98E-05	4.08E-07
106-110	1.90E-05	1.95E-05	1.82E-05	1.89E-05	6.68E-07
116-125	2.11E-05	2.14E-05	2.05E-05	2.10E-05	4.98E-07
126-138	2.14E-05	2.18E-05	2.09E-05	2.13E-05	4.64E-07
139-143	1.63E-05	1.66E-05	1.58E-05	1.62E-05	3.87E-07
146-150	2.03E-05	2.06E-05	1.94E-05	2.01E-05	6.20E-07
151-169	1.95E-05	2.00E-05	1.91E-05	1.95E-05	4.37E-07

#### 4.4.3 A59G Mutant (PDB ID 6ASE)

KRAS operates through a dynamic interplay between two states: an active GTP-bound state and an inactive GDP-bound state. The A59G mutation, as performed by Lu, et al., [18], was strategically designed to "lock" KRAS into the inactive GDP-bound state. The transition to this inactive starting structure appears to induce a noteworthy shift in the overall diffusion characteristics of the protein. This transition is not confined to isolated regions but instead manifests across the entire protein, suggesting an impact on its behavior. The alteration in the conformation of the switch regions, coupled with an augmented flexibility observed in residues 57 to 61, could plausibly be contributing factors to the observed increase in hydration dynamics.

Table 13: Diffusion Coefficients for A59G Mutant (PDB ID 6ASE)

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	A59G Run 1	A59G Run 2	A59G Run 3	Average	SD
1-9	1.77E-05	1.92E-05	1.82E-05	1.86E-05	7.80E-07
10-14	1.58E-05	1.47E-05	1.40E-05	1.48E-05	7.38E-07
15-26	1.61E-05	1.71E-05	1.66E-05	1.68E-05	6.15E-07
27-37	2.12E-05	2.08E-05	2.04E-05	2.10E-05	5.14E-07
38-50	1.93E-05	2.08E-05	1.99E-05	2.03E-05	9.32E-07
56-63	1.68E-05	1.67E-05	1.60E-05	1.67E-05	5.37E-07
64-75	1.99E-05	2.12E-05	2.04E-05	2.07E-05	6.38E-07
86-105	1.98E-05	2.03E-05	1.95E-05	2.00E-05	4.26E-07
106-110	1.86E-05	1.84E-05	1.84E-05	1.84E-05	2.02E-07
116-125	2.11E-05	2.14E-05	2.08E-05	2.12E-05	3.41E-07
126-138	2.11E-05	2.15E-05	2.06E-05	2.13E-05	5.66E-07
139-143	1.59E-05	1.63E-05	1.59E-05	1.62E-05	5.21E-07
146-150	2.04E-05	2.07E-05	2.03E-05	2.05E-05	1.50E-07
151-169	1.92E-05	1.98E-05	1.87E-05	1.95E-05	6.57E-07

#### 4.4.4 A59G I36T and A59G M67T Double Mutants

When we replicate the same mutations in the "locked" close state, which mimics the GDP-bound conformation of KRAS, intriguing insights emerge. These findings parallel the previously mentioned observations by Franck, where water dynamics exhibited an increase around residue 36 and a decrease around residue 67. This consistency in results across different contexts is noteworthy and warrants further exploration. One possible explanation for these recurring patterns lies in the fact that, despite the absence of GDP binding, the protein remains essentially "locked" in a configuration that closely resembles the bound state. This state of confinement within a GDP-like position may have a profound impact on the dynamic behavior of the protein and its interaction with surrounding water molecules. As a result, the dynamics surrounding residues 36 and 67 may behave as if GDP were indeed bound, illustrating the intricate relationship between the structural state of KRAS and its hydration dynamics. These findings underscore the complexity of the KRAS system and emphasize the importance of

considering the specific structural context when interpreting hydration dynamics data. Further investigations into the underlying mechanisms driving these observations are warranted to gain a comprehensive understanding of how mutations influence KRAS behavior in different conformational states.

Table 14: Diffusion Coefficients for A59G I36T Double Mutant

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	A59G I36T Run 1	A59G I36T Run 2	A59G I36T Run 3	Average	SD
1-9	1.71E-05	1.65E-05	1.71E-05	1.69E-05	3.67E-07
10-14	1.21E-05	1.25E-05	1.27E-05	1.24E-05	3.11E-07
15-26	1.43E-05	1.53E-05	1.51E-05	1.49E-05	4.91E-07
27-37	1.91E-05	1.89E-05	1.93E-05	1.91E-05	2.25E-07
38-50	1.89E-05	1.86E-05	1.90E-05	1.88E-05	1.82E-07
56-63	1.49E-05	1.49E-05	1.53E-05	1.50E-05	2.57E-07
64-75	1.89E-05	1.93E-05	1.91E-05	1.91E-05	1.97E-07
86-105	1.81E-05	1.82E-05	1.83E-05	1.82E-05	7.71E-08
106-110	1.70E-05	1.61E-05	1.62E-05	1.64E-05	5.26E-07
116-125	1.90E-05	1.90E-05	1.93E-05	1.91E-05	2.12E-07
126-138	1.95E-05	1.94E-05	1.95E-05	1.95E-05	7.38E-08
139-143	1.42E-05	1.43E-05	1.42E-05	1.43E-05	6.47E-08
146-150	1.73E-05	1.79E-05	1.72E-05	1.75E-05	3.91E-07
151-169	1.78E-05	1.78E-05	1.78E-05	1.78E-05	3.32E-08

Table 15: Diffusion Coefficients for A59G M67T Double Mutant

Diffusion Coefficients cm <sup>2</sup> /ps					
Residues	A59G M67T Run 1	A59G M67T Run 2	A59G M67T Run 3	Average	SD
1-9	1.70E-05	1.73E-05	1.72E-05	1.72E-05	1.46E-07
10-14	1.39E-05	1.27E-05	1.25E-05	1.31E-05	7.84E-07
15-26	1.44E-05	1.45E-05	1.43E-05	1.44E-05	6.91E-08
27-37	2.01E-05	1.87E-05	1.92E-05	1.93E-05	7.39E-07
38-50	1.81E-05	1.86E-05	1.86E-05	1.84E-05	3.10E-07
56-63	1.63E-05	1.59E-05	1.50E-05	1.58E-05	6.97E-07
64-75	1.91E-05	1.87E-05	1.90E-05	1.89E-05	2.11E-07
86-105	1.86E-05	1.84E-05	1.85E-05	1.85E-05	9.10E-08
106-110	1.65E-05	1.66E-05	1.64E-05	1.65E-05	1.06E-07
116-125	1.96E-05	1.92E-05	1.94E-05	1.94E-05	2.14E-07
126-138	1.96E-05	1.96E-05	1.97E-05	1.96E-05	4.44E-08
139-143	1.43E-05	1.42E-05	1.46E-05	1.43E-05	1.88E-07
146-150	1.78E-05	1.84E-05	1.84E-05	1.82E-05	3.29E-07
151-169	1.77E-05	1.75E-05	1.76E-05	1.76E-05	1.06E-07

#### 4.4.5 Diffusion Coefficient Summary

The discernible impact of each mutation on the overarching hydration landscape of KRAS is readily apparent when we refer to Table 22 and Figure 24. It is worth noting that the influence of each mutation on the hydration layer becomes notably pronounced and statistically significant after the introduction of just two mutations. This striking pattern underscores the remarkable tunability of KRAS and serves as a compelling testament to its sensitivity to even alterations at a single residue. These observations illuminate the intricacies of KRAS dynamics and its remarkable responsiveness to perturbations. They emphasize that subtle changes within the protein's structure, such as those induced by single-point mutations, ripple through the entire protein environment, shaping its hydration properties. This inherent sensitivity to modifications highlights the complexity of KRAS and underscores the need for a nuanced understanding of its behavior and the potential implications of mutations in the context of cellular signaling.

pathways. Further exploration and analysis are warranted to unravel the full extent of KRAS's adaptability and its role in intracellular signaling.

Table 16: Diffusion Coefficient of all KRAS structures

KRAS Average Diffusion Coefficients cm <sup>2</sup> /s							
	Residues	WT	I36T	M67T	A59G	A59G I36T	A59G M67T
1	1-9	1.78E-05	1.80E-05	1.85E-05	1.86E-05	1.69E-05	1.72E-05
2	10-14	1.34E-05	1.40E-05	1.46E-05	1.48E-05	1.24E-05	1.31E-05
3	15-26	1.65E-05	1.68E-05	1.77E-05	1.68E-05	1.49E-05	1.44E-05
4	27-37	1.95E-05	2.05E-05	2.06E-05	2.10E-05	1.91E-05	1.93E-05
5	38-50	1.98E-05	2.01E-05	2.06E-05	2.03E-05	1.88E-05	1.84E-05
6	56-63	1.58E-05	1.66E-05	1.66E-05	1.67E-05	1.50E-05	1.58E-05
7	64-75	1.97E-05	2.05E-05	2.08E-05	2.07E-05	1.91E-05	1.89E-05
8	86-105	1.92E-05	1.97E-05	1.98E-05	2.00E-05	1.82E-05	1.85E-05
9	106-110	1.76E-05	1.83E-05	1.89E-05	1.84E-05	1.64E-05	1.65E-05
10	116-125	1.99E-05	2.09E-05	2.10E-05	2.12E-05	1.91E-05	1.94E-05
11	126-138	2.05E-05	2.11E-05	2.13E-05	2.13E-05	1.95E-05	1.96E-05
12	139-143	1.50E-05	1.62E-05	1.62E-05	1.62E-05	1.43E-05	1.43E-05
13	146-150	1.89E-05	1.99E-05	2.01E-05	2.05E-05	1.75E-05	1.82E-05
14	151-169	1.85E-05	1.94E-05	1.95E-05	1.95E-05	1.78E-05	1.76E-05

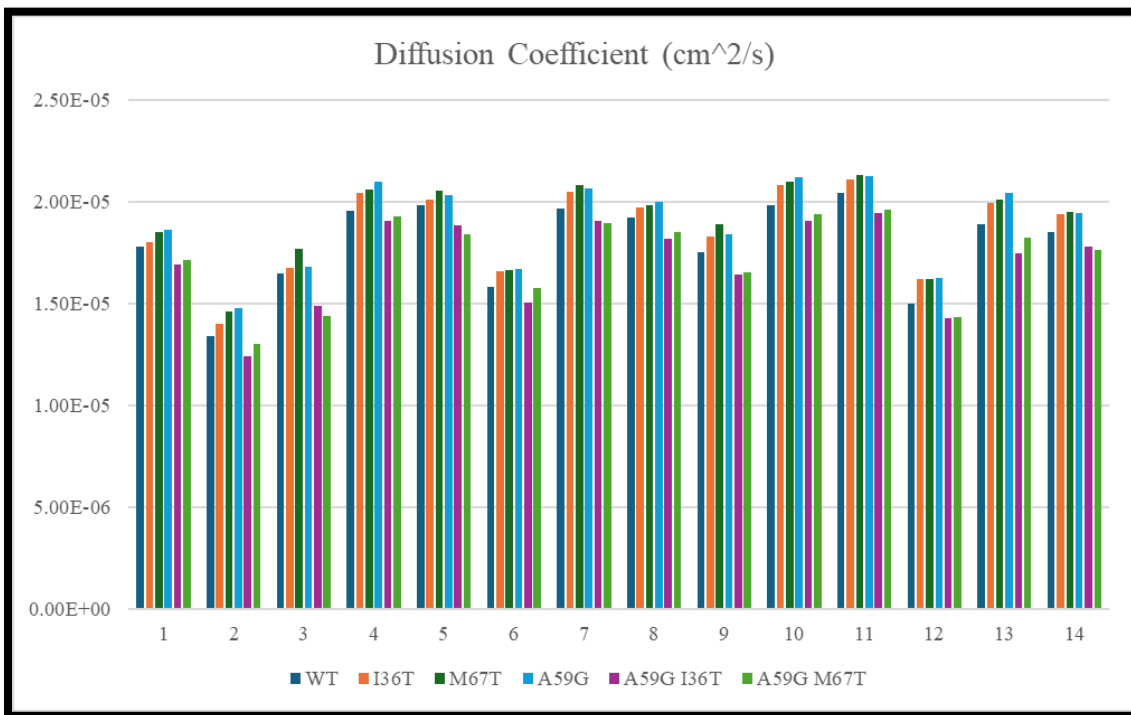


Figure 23: Comparison of Diffusion Coefficients

## CHAPTER 5

### CONCLUSIONS AND FUTURE DIRECTIONS

#### 5.1 Conclusions

Exploring and gaining a comprehensive understanding of the hydration shell surrounding proteins is an essential key to unraveling the intricacies of protein structure and function. This study has shed light on the presence of two distinct solvation shell environments and has demonstrated how these environments are influenced by alterations in bound peptides or mutations at specific residues. In the case of MHC proteins, which play a crucial role in molecular recognition, the conserved nature of the hydration layer underscores its fundamental importance. Maintaining a consistent hydration landscape is essential for their ability to effectively present peptides, as excessive variability could hinder their functional role. The sensitivity of KRAS to even a single mutation within its structure, as evidenced by changes in hydration shell dynamics, is indicative of its finely tuned molecular machinery. This heightened sensitivity may be a key factor in understanding how mutated KRAS can undergo functional changes that contribute to cancer development. As we continue to delve deeper into the complexities of protein behavior, it becomes increasingly clear that a more profound understanding of the hydration shell is indispensable. Furthermore, advancements in our comprehension of the hydration shell are pivotal in our ongoing exploration of protein structure and function. The development of tools like the Fast Analysis Suite represents a significant leap forward in this endeavor, offering researchers across the globe a streamlined and reproducible means to investigate molecular dynamics through simulations. These tools empower researchers

to delve into the dynamic world of proteins and further our understanding of their roles in cellular processes and diseases.

## REFERENCES

## REFERENCES

1. Fenimore, P. W.; Frauenfelder, H.; McMahon, B. H.; Young, R. D. Bulk-Solvent and Hydration-Shell Fluctuations, Similar to  $\alpha$ - and  $\beta$ -Fluctuations in Glasses, Control Protein Motions and Functions. *Proc. Natl. Acad. Sci. U.S.A.* **2004**, *101* (40), 14408–14413. <https://doi.org/10.1073/pnas.0405573101>.
2. Frauenfelder, H.; Fenimore, P. W.; Chen, G.; McMahon, B. H. Protein Folding Is Slaved to Solvent Motions. *Proc. Natl. Acad. Sci. U.S.A.* **2006**, *103* (42), 15469–15472. <https://doi.org/10.1073/pnas.0607168103>.
3. Fogarty, A. C.; Laage, D. Water Dynamics in Protein Hydration Shells: The Molecular Origins of the Dynamical Perturbation. *J. Phys. Chem. B* **2014**, *118* (28), 7715–7729. <https://doi.org/10.1021/jp409805p>.
4. Dahanayake, J. N.; Mitchell-Koch, K. R. How Does Solvation Layer Mobility Affect Protein Structural Dynamics? *Front. Mol. Biosci.* **2018**, *5*, 65. <https://doi.org/10.3389/fmolb.2018.00065>.
5. Ali, S.; Hassan, Md.; Islam, A.; Ahmad, F. A Review of Methods Available to Estimate Solvent-Accessible Surface Areas of Soluble Proteins in the Folded and Unfolded States. *CPPS* **2014**, *15* (5), 456–476. <https://doi.org/10.2174/1389203715666140327114232>.
6. Durham, E.; Dorr, B.; Woetzel, N.; Staritzbichler, R.; Meiler, J. Solvent Accessible Surface Area Approximations for Rapid and Accurate Protein Structure Prediction. *J Mol Model* **2009**, *15* (9), 1093–1108. <https://doi.org/10.1007/s00894-009-0454-9>.
7. Tan, J. M. M.; Wong, E. S. P.; Lim, K.-L. Protein Misfolding and Aggregation in Parkinson's Disease. *Antioxidants & Redox Signaling* **2009**, *11* (9), 2119–2134. <https://doi.org/10.1089/ars.2009.2490>.
8. Li, K.; Tokareva, O. S.; Thomson, T. M.; Wahl, S. C. T.; Travaline, T. L.; Ramirez, J. D.; Choudary, S. K.; Agarwal, S.; Walkup, W. G.; Olsen, T. J.; Brennan, M. J.; Verdine, G. L.; McGee, J. H. De Novo Mapping of  $\alpha$ -Helix Recognition Sites on Protein Surfaces Using Unbiased Libraries. *Proc. Natl. Acad. Sci. U.S.A.* **2022**, *119* (52), e2210435119. <https://doi.org/10.1073/pnas.2210435119>.
9. Frenkel, D.; Smit, B. *Understanding Molecular Simulation: From Algorithms to Applications*, 2nd ed.; Computational science series; Academic Press: San Diego, 2002.
10. Hewitt, E. W. The MHC Class I Antigen Presentation Pathway: Strategies for Viral Immune Evasion. *Immunology* **2003**, *110* (2), 163–169. <https://doi.org/10.1046/j.1365-2567.2003.01738.x>.

11. Wieczorek, M.; Abualrous, E. T.; Sticht, J.; Álvaro-Benito, M.; Stolzenberg, S.; Noé, F.; Freund, C. Major Histocompatibility Complex (MHC) Class I and MHC Class II Proteins: Conformational Plasticity in Antigen Presentation. *Front. Immunol.* **2017**, *8*. <https://doi.org/10.3389/fimmu.2017.00292>.
12. Ayres, C. M.; Riley, T. P.; Corcelli, S. A.; Baker, B. M. Modeling Sequence-Dependent Peptide Fluctuations in Immunologic Recognition. *J. Chem. Inf. Model.* **2017**, *57* (8), 1990–1998. <https://doi.org/10.1021/acs.jcim.7b00118>.
13. Mark, P.; Nilsson, L. Structure and Dynamics of the TIP3P, SPC, and SPC/E Water Models at 298 K. *J. Phys. Chem. A* **2001**, *105* (43), 9954–9960. <https://doi.org/10.1021/jp003020w>.
14. Laage, D.; Elsaesser, T.; Hynes, J. T. Water Dynamics in the Hydration Shells of Biomolecules. *Chem. Rev.* **2017**, *117* (16), 10694–10725. <https://doi.org/10.1021/acs.chemrev.6b00765>.
15. Stolze, B.; Reinhart, S.; Bullinger, L.; Fröhling, S.; Scholl, C. Comparative Analysis of KRAS Codon 12, 13, 18, 61 and 117 Mutations Using Human MCF10A Isogenic Cell Lines. *Sci Rep* **2015**, *5* (1), 8535. <https://doi.org/10.1038/srep08535>.
16. Syed, F.; Khuc, J. N.; Guinness, A.; Franck, J. M. Contiguous Patches of Translational Hydration Dynamics on the Surface of K-Ras. arXiv July 10, 2023. <http://arxiv.org/abs/2307.03179> (accessed 2024-01-02).
17. Franck, J. M.; Pavlova, A.; Scott, J. A.; Han, S. Quantitative Cw Overhauser Effect Dynamic Nuclear Polarization for the Analysis of Local Water Dynamics. *Progress in Nuclear Magnetic Resonance Spectroscopy* **2013**, *74*, 33–56. <https://doi.org/10.1016/j.pnmrs.2013.06.001>.
18. Lu, J.; Bera, A. K.; Gondi, S.; Westover, K. D. KRAS Switch Mutants D33E and A59G Crystallize in the State 1 Conformation. *Biochemistry* **2018**, *57* (3), 324–333. <https://doi.org/10.1021/acs.biochem.7b00974>.
19. Abraham, M.; Alekseenko, A.; Basov, V.; Bergh, C.; Briand, E.; Brown, A.; Doijade, M.; Fiorin, G.; Fleischmann, S.; Gorelov, S.; Gouaillardet, G.; Grey, A.; Irrgang, M. E.; Jalalypour, F.; Jordan, J.; Kutzner, C.; Lemkul, J. A.; Lundborg, M.; Merz, P.; Miletic, V.; Morozov, D.; Nabet, J.; Pall, S.; Pasquadibisceglie, A.; Pellegrino, M.; Santuz, H.; Schulz, R.; Shugaeva, T.; Shvetsov, A.; Villa, A.; Wingbermuehle, S.; Hess, B.; Lindahl, E. GROMACS 2024.1 Manual. **2024**. <https://doi.org/10.5281/ZENODO.10721192>.