Exploring Protein Phase Space using Machine Learning based Enhanced Sampling

A Thesis

submitted to
Indian Institute of Science Education and Research Pune
in partial fulfillment of the requirements for the
BS-MS Dual Degree Programme

by

Gopi Krishnan R

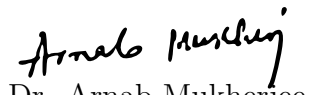
Indian Institute of Science Education and Research Pune
Dr. Homi Bhabha Road,
Pashan, Pune 411008, INDIA.

December, 2022

Supervisor: Dr. Arnab Mukherjee

Certificate

This is to certify that this dissertation entitled Exploring Protein Phase Space using Machine Learning based Enhanced Samplingtowards the partial fulfilment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research, Pune represents study/work carried out by Gopi Krishnan Rat Indian Institute of Science Education and Research under the supervision of Dr. Arnab Mukherjee, Professor, Department of Chemistry , during the academic year 2021-2022.



Dr. Arnab Mukherjee

Committee:

Dr. Arnab Mukherjee

Dr. M. S. Madhusudhan

This thesis is dedicated to Dr. Arnab Mukherjee and all my lab mates.

Declaration

I hereby declare that the matter embodied in the report entitled Exploring Protein Phase Space using Machine Learning based Enhanced Sampling are the results of the work carried out by me at the Department of Chemistry, Indian Institute of Science Education and Research, Pune, under the supervision of Dr. Arnab Mukherjee and the same has not been submitted elsewhere for any other degree.

Gopi Krishnan R

Acknowledgments

I would like to thank Dr. Arnab Mukherjee for his invaluable support throughout this project. I would also like to thank my fellow lab mates, whose suggestions helped me complete this project faster. I would also like to thank Dr. Prabhakar Bhimalapuram, Assistant Proffessor, IIIT Hyderabad and his student. This idea was initially proposed by them and we worked on this idea and tried to expand it for a larger system, also adding few modifications on the way. Finally I would like to thank IISER Pune for providing a HPC like Param Brahma, which made my simulation and training much faster than initially expected.

Abstract

Molecular Dynamics simulations of Condensed matter systems are known to have very slow dynamics and thus often leading to poor sampling of the phase space. This can lead to wrong interpretation of system's macroscopic properties properties. In this paper, we would like to introduce a machine learning based approach that can efficiently explore the *phase space* of a system. This is done in a two-step process: (1) classify metastable states into unique clusters (classifier model) (ii) develop a non-linear relation within each cluster (regressor model). We apply this method on a simple system like alanine tetrapeptide and show that there is a drastic improvement in sampling compared to a benchmark study with already established collective variables. We also apply this method on a larger peptide sequence, namely a Chameleon sequence. The model is able to reach all the secondary structures with reasonable sampling.

Contents

Abstract **xi**

1 Introduction **1**

2 Theory **3**

2.1 Molecular Dynamics 3

2.2 Enhanced Sampling 3

2.3 Ramachandran plot 5

2.4 Machine Learning 5

3 Methods **7**

3.1 Machine Learning Models 7

3.2 Algorithm 10

3.3 System 11

4 Results and Discussion **15**

4.1 Alanine Tetrapeptide 15

4.2 Chameleon Sequences 18

5 Conclusion and Future Outlooks **23**

List of Figures

2.1 Molecular dynamics and Enhanced Sampling 4

3.1 Machine Learning Models 8

3.2 Algorithm 10

3.3 Alanine tetrapeptide 12

4.1 Plot of Time spent in each Metastable state 16

4.2 Graph Diagram of Metastable states 17

4.3 Free Energy Plot for Alanine tetrapeptide 18

4.4 Plot of Secondary Structure content 20

4.5 Various Secondary Structures 21

4.6 Ramachandran Plot 22

5.1 2-D Free energy plot of Chameleon sequence 24

Chapter 1

Introduction

Molecular Dynamics (MD) is a powerful tool for studying the dynamics of a system. But for a realistic system like Protein, the free energy landscape is often complicated. Thus MD alone is not that useful as it gets stuck to local minima, which we call a metastable state. One must sacrifice the dynamics by accelerating the sampling to sample the entire phase space. One common way is to use Enhanced Sampling, which adds periodic biasing potential to the system and accelerates the process of barrier crossing. Many such techniques rely on probing the probability distribution along a selected direction or directions, which we call " Collective Variables." The sampled distribution's quality and reliability heavily depend on the information in the chosen collective variable. Mainly, there are some essential criteria for selecting a collective variable - a) It should be able to differentiate between the relevant metastable state, b) It should be able to identify the limiting dynamics and thus accelerate it, c) It should be low dimensional, ideally not more than two. In most cases, there is no obvious way to get an excellent collective variable. But one needs to find it to perform enhanced sampling. Thus the general approach to deal with this issue is,

1. Using Intuitive Collective Variable - here we choose the collective variables specific to the system, often chosen on a trail and error basis. eg: end-to-end distance for a protein folding/unfolding process. It is not easy to find an intuitive collective variable for a large system.

2. Post-processing the trajectory - Here we rely on an intuitive CV to sample the phase space and then post-process the trajectory to get a new CV that can sample the phase

space much better than the intuitive CV. eg. performing PCA onto the trajectory[1]. Some of the notable contribution in this approach include SGOOP[2], SGOOP-d[3], TICA[4], Deep-TICA[5], Deep-LDA[6]

Dr. Prabhakar Bhimalapuram, Assistant Professor IIIT Hyderabad, proposed an alternative method that learns the collective variable from the trajectory but does not rely on the intuitive collective variable to sample the dynamics first. This is achieved by combining the power of "Unsupervised Learning" and Enhanced Sampling. Rather than relying on an intuitive collective variable, the method uses the learned parameter of the model as the CV and performs enhanced sampling along its direction. This thesis is an extension of his work, trying to apply the method to a more extensive system and adding some modifications. This modification helps us get helpful information like free energy from it, which was previously not possible. We applied this method on two systems - a) Alanine tetrapeptide and b) a 16-residue Chameleon sequence. Dr. Prabhakar Bhimalapuram used Alanine tetrapeptide as a model system in his work. We used the same as a starting point for validating the extension. To validate the method, we also compared it with two widely used ways - TICA and SGOOP-d. We show that our method performs equally or slightly better (in some instances) with much lower simulation time.

Chapter 2

Theory

2.1 Molecular Dynamics

Molecular Dynamics (MD) simulations were first developed in the late 1970s. Before this, calculations were required to provide quantum-mechanical motions and proved to be computationally intensive for the best of supercomputers. MD simulations seek to overcome this by using Newton's laws-based approximations to simulate atomic motions. The general process of this approximation is shown in figure 2.1(a). We start with an initial structure, usually prepared from NMR, crystallographic, or homology modeling. We then calculate the forces acting on each atom. Forces arise from Bonded and Non-bonded interactions. Bonds and angles are modeled using simple springs, while dihedrals are modeled using the sinusoidal function. Non-bonded interactions arise from charged interactions, modeled using Coulomb's law, and van der Waals interaction, modeled using Lennard-Jones potential.

2.2 Enhanced Sampling

In most natural systems, Molecular Dynamics alone cannot sample the relevant phase space. Most rare events occur in a minimum time scale of microseconds, and performing molecular dynamics for such a prolonged duration is computationally inefficient. Thus one has to compromise on the natural dynamics of the system for better sampling. Enhanced sampling

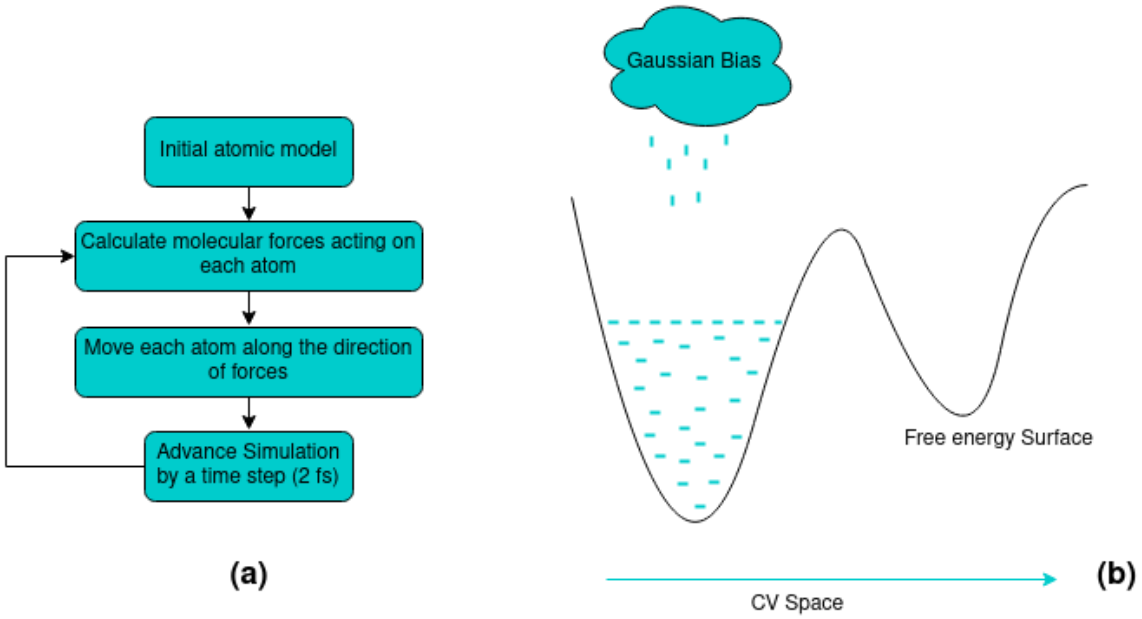


Figure 2.1: a) This shows the general procedure for a molecular dynamics simulation, b) A pictorial representation of adding of bias potential into a well.

is one such approach. Among the various Enhanced Sampling techniques, one widely used is Metadynamics[7]. Here we add history-dependent Gaussian potentials to push the system from a metastable state to another figure 2.1(b). In this paper, we will use a form of Metadynamics known as Well-Tempered Metadynamics[8]. The general equation of Metadynamics is given as follows,

$$V_b(s,t) = \sum_{t=\tau_G,2\tau_G,..} w\ exp\left(-\frac{|s-s(t)|^2}{2\sigma^2}\right) \tag{2.1}$$

where s is a collective variable vector σ is the width and w is the height of the Gaussian. This potential fills up the wells in the free energy surface, and the system escapes from the metastable state. Once all of the states have been explored, the bias potential will give us the free energy, $F(s) = -V_b(s, t \to \infty)$. The error associated with the free energy estimate depends on W, σ and τ_G. Once the complete free energy is explored, additional bias will introduce a fluctuation of the free energy value around the actual. Thus one has to control this fluctuations depending on the added bias.

One of the promising attempt is Well-Tempered approach. Here we control the height of

the added bias as given by the following relation

$$w = w_0 e^{-[V_b(s,t)/\Delta T]} \tau_G \tag{2.2}$$

where, w_0 is the deposition rate, $V_b(s,t)$ is the bias potential and Δ T is a hyper-parameter. We can get the free energy estimate from the bias potential, and the scaling factor used is called as BIAS-FACTOR.

$$F(s) = -\frac{T + \Delta T}{\Delta T} V_b(s, t \to \infty) \tag{2.3}$$

Initially, the potential will be zero and thus w will be at it maximum, this helps in faster filling of the wells. Eventually, w gets scaled and thus only small changes are made to the potential, leading to a smooth convergence of the actual free energy surface. We will choose the values of the ω and ΔT for the best efficiency.

2.3 Ramachandran plot

Ramachandran plot[9] is a 2D plot of ϕ and ψ torsional angles of amino acid residues in a protein or peptide. The ϕ represents the angle between $N_{(i-1)} - C_{(i)} - CA_{(i)} - N_{(i)}$ and ψ represents the angle between $C_{(i)} - CA_{(i)} - N_{(i)} - C_{(i-1)}$. Plotting the torsional angles in this manner will graphically shows us which combination of ϕ and ψ are sterically possible. By making a ramachandran plot for a protein, one can obtain insight into the secondary structure of the protein.

2.4 Machine Learning

Machine Learning is an approach where one allows the computer to learn without being explicitly programmed. Traditional programming relies on predefined fixed instructions, whereas machine learning can find the instruction given the data and outcome. Among all the machine learning techniques, "Unsupervised Learning" fascinates us. As the name suggests, unsupervised learning can capture information from unlabeled data. The Molecular dynamics trajectory is mostly unlabeled in phase space, and thus we can use it to capture the

essential dynamics of the system. In this study, we are utilizing one of the widely used Unsupervised models, named Autoencoder[10]. This model can learn to encode high-dimensional data into a low-dimensional representation. We will then use this low dimensional representation to bias the system to its direction and ideally hope to improve phase space sampling.

The quality of inputs plays a massive role in the information learned by the model. Ideally, one would need an input trajectory containing information about the entire phase space. However, one needs an excellent collective variable to sample the phase space. This essentially boils down to the chicken and egg problem. In order to solve this issue, we use a classification model that divides the phase space into small clusters and trains an individual model. Thus rather than trying to find out a collective variable for the entire trajectory, we learn the collective variable for each metastable state and find a path through the minimum free energy.

Chapter 3

Methods

In this section we provide detailed explanation about, 1) Machine Learning Models, 2) Algorithm and 3) System used.

3.1 Machine Learning Models

We use Machine learning models to identify the collective variable. The proposed method utilizes both an unsupervised classifier and a regressor. Since the learning is unsupervised, an apt regressor would be an Auto-associative Artificial Neural Network, commonly known as "auto-encoders." For the classification, we will use a clustering classifier called the Elliptic Envelope scheme[11], which is a reasonable choice when your distribution is Gaussian.

Regressor Model

Regressor models tell us the relationship between an independent variable quantified to one or more dependent variables. It helps us to sort out the essential dependent variables. Auto-encoders are one such network whose primary goal is to reduce the dimension of the input data and reproduce the data from the reduced dimension. A fully trained Autoencoder will have a reduced dimension space (called "bottleneck layer") encoded with the maximum information about the input data. A general structure of the auto-encoder is shown in figure

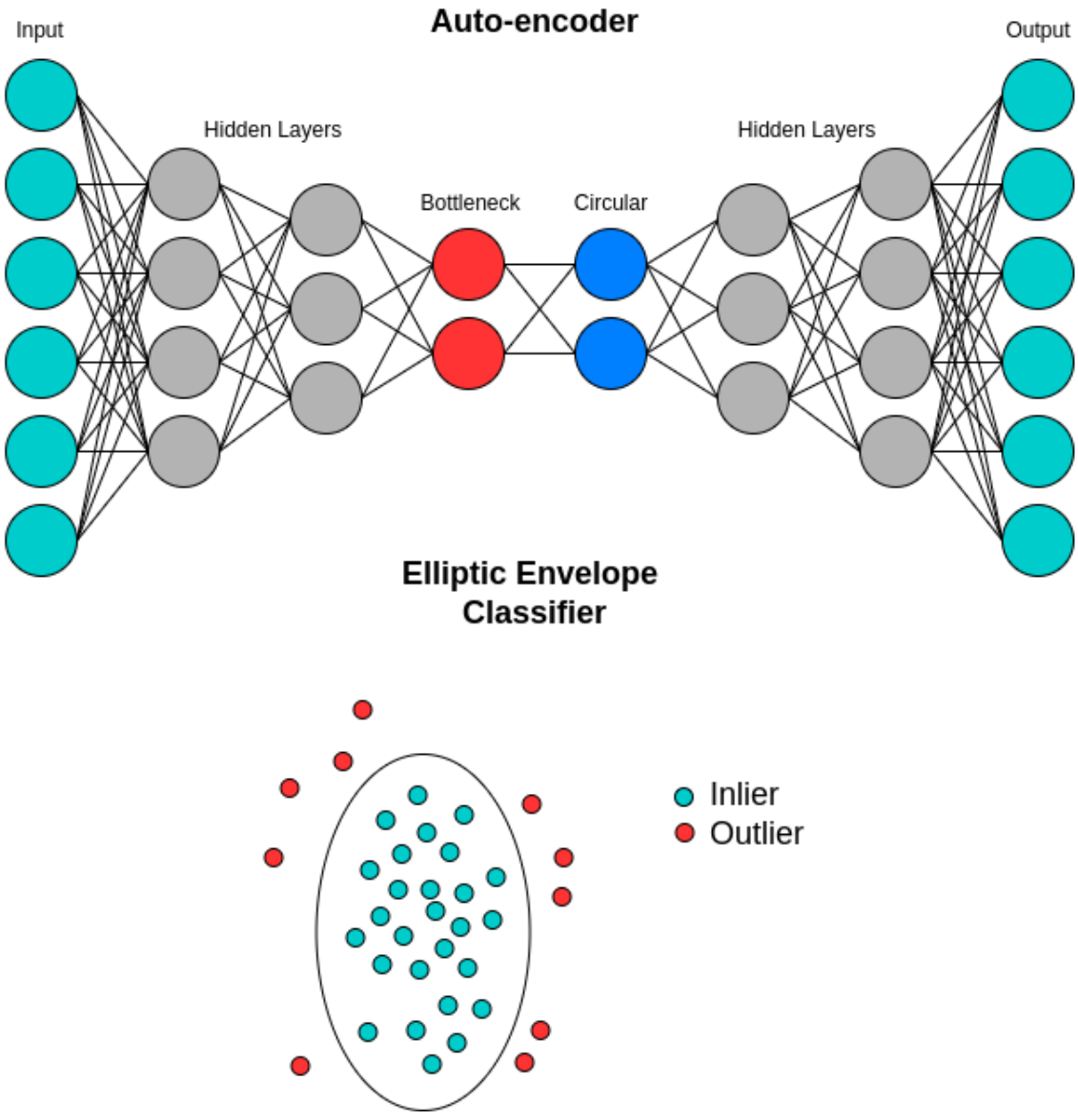


Figure 3.1: The figure shows the two different types of models used in this method.

3.1(a). The auto-encoder structure utilizes a circular projection, following Ferguson groups' work[12].

Each node takes a bias and a weighted sum of outputs from the nodes in the previous layer and generates its output via an activation function. Here the activation function used is a hyperbolic tangent function. The non-linear activation function makes the model more complex and thus helps the model to learn more complex relationships. Apart them that, they also ensure that the underlying functional relation is continuous and possesses well-defined gradients. We train the network by iterating over a series of examples with a loss function that minimizes the reconstruction error. The training is performed via a mini-batch stochastic gradient descent for 500 epochs with Adam as the Optimizer with a learning rate of 1e-4. To prevent over-fitting, we use a train-test split of 80:20. The two dimension bottleneck layer of the encoder is used for the Metadynamics. The input layer of the model is the sine and cosine of the ϕ and ψ dihedral angles centered around their mean value.

Classifier Model

Classifier models tell us a qualitative relation between an independent variable and one or more dependent variables. These are usually some clustering algorithm that tells us the probability of the given data point belonging to a particular cluster. We can apply the clustering directly on the input data or at the latent dimension. The clustering algorithm uses outlier detection, clusters the region where training data is mainly located with a solid boundary and demarcates the outlier data points. The Classifier model used here is the "Elliptic Envelope Scheme." This classifier uses an ellipse as the solid boundary and assumes that the inliers are from the same Gaussian distribution. This method works well for metastable states and degrades when the data deviates from a unimodal distribution. Also, the classier is designed to have an inherent preference towards the initially found classes. The classes converge to a few significant clusters (ideally one). Thus we can use the latent representation of that class(s) as the collective variable for efficient sampling.

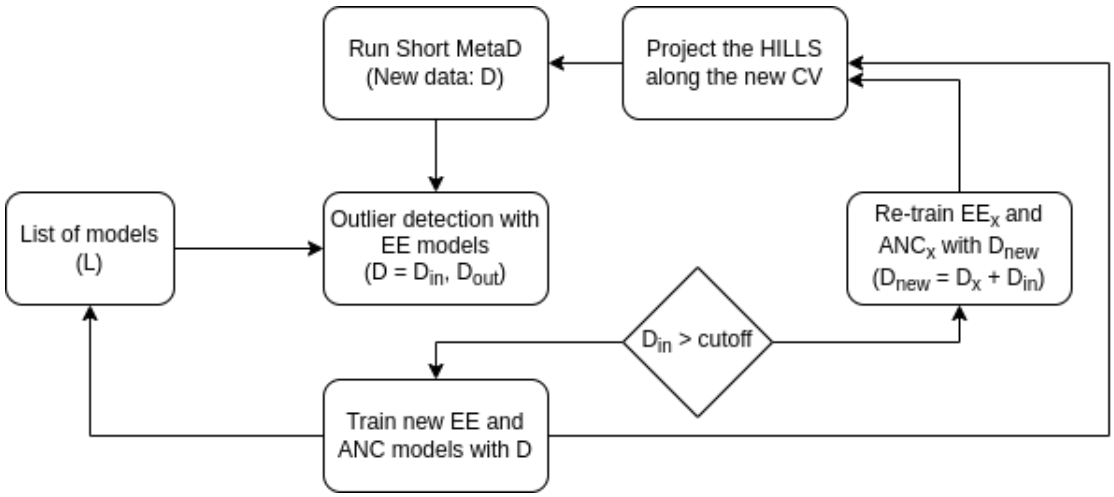


Figure 3.2: The given flow chart explains the algorithm used in this protocol.

3.2 Algorithm

1. We start with the data from 10 ns unbiased simulation. The sine and cosine of the re-centered dihedral will be our input data.

2. We will perform classification on it and then a regressor is trained for the classified cluster. Initially we don't have any formed clusters, thus the whole data will be a single cluster.

3. Once the regression is done we will use the latent layer of the model to run a short 1 ns well-tempered metadynamics, this will generate new data (D). We will also re-weight this trajectory and save the bias added for later use (free energy calculation).

4. The newly generated data (D) will be compared with data in each cluster, this will result in inliers (D_{in}) and outliers (D_{out}) and the classification is done as follows :

   - If we find a cluster with D_{in} greater than a predefined cut-off of D (we choose, 60%), then the data set of first such cluster is updated as $D_x = D_x + D_{in}$ (where x is the cluster number) and the regressor for that cluster is re-trained.
   - If D_{in} is less than the predefined cut-off of D for all the clusters, then we will create a new cluster $D_{x+1} = D$, and a new regressor is trained.

5. Now we will rerun the metadynamics along the latent layer of the model found in step 4, this will give us a HILLS file projected along the new CV.

We will then run an extended 1 ns well-tempered metadynamics with the new CV. Step 3, 4 and 5 are repeated until the simulation converges along a fixed predefined collective variable. The classification has a preference towards the initially found clusters. This helps us in getting dominant clusters and reduce the chance of similar clusters.

3.3 System

The Simulations were performed in Gromacs 2021.4[13] [14] patched with plumed 2.7.4[15] (enabling the optional ANN module). For all the systems we used the force field AMBER-ff99SB[16]. Temperature was kept constant at 300K and velocity re-scale thermostat was used. A time-step of 2 fs was used and all bonds involving hydrogen were constrained. For enhanced Sampling we will be using Well-Tempered Metadynamics (WT-MetaD).

Alanine Tetrapeptide

Alanine tetrapeptide (ACE-$(ALA)_3$-NME) is the first model system used to validate the method. Despite being a toy system, it has a relatively complicated free energy surface which the intuitive collective variable struggles to sample fully, thus making it an ideal toy system to test the model's performance. The change in dihedral angles distinguishes the metastable states; therefore, we will be using it as the basis for the construction of CV. We will use all six backbone dihedral angles (see the below figure). The input for the regressor model is the re-centered (i.e., mean subtraction) sine and cosine of dihedral angles. The structure of the Autoencoder model used here is :

$$12 - 8 - 4 - 2(Bottleneck) - 2(1Dcircular) - 4 - 8 - 12$$

The bottleneck layer coordinates are used as CV for WT-MetaD simulations. In order to study the explored states and configurations, we will divide the phase space of the molecule. We can have two useful division schemes based on the dihedral angles (ϕ and ψ). First division is based on ϕ alone, where we look for the sign of ϕ, ie. $phi > 0$ or $phi < 0$ and we will give them a label 0_2 and 1_2. The second division is based on both ϕ and ψ; this divides the free energy landscape into four regions, and we label them as $0_4, 1_4, 2_4, 3_4$.

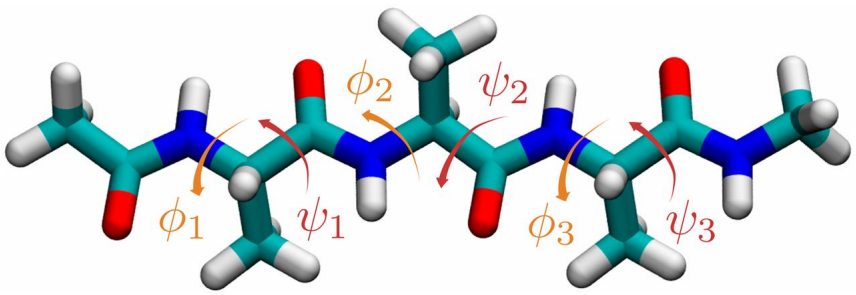


Figure 3.3: The figure shows the ϕ and ψ angles of alanine tetrapeptide.

State	Binary	Binary to Decimal	Metastate
$(0_2, 0_2, 0_2)$	000_2	$[0 * 2^2 + 0 * 2^1 + 0 * 2^0]$	0
$(0_2, 0_2, 1_2)$	001_2	$[0 * 2^2 + 0 * 2^1 + 1 * 2^0]$	1
$(0_2, 1_2, 0_2)$	010_2	$[0 * 2^2 + 1 * 2^1 + 0 * 2^0]$	2
$(0_2, 1_2, 1_2)$	011_2	$[0 * 2^2 + 1 * 2^1 + 1 * 2^0]$	3
$(1_2, 0_2, 0_2)$	100_2	$[1 * 2^2 + 0 * 2^1 + 0 * 2^0]$	4
$(1_2, 0_2, 1_2)$	101_2	$[1 * 2^2 + 0 * 2^1 + 1 * 2^0]$	5
$(1_2, 1_2, 0_2)$	110_2	$[1 * 2^2 + 1 * 2^1 + 0 * 2^0]$	6
$(1_2, 1_2, 1_2)$	111_2	$[1 * 2^2 + 1 * 2^1 + 1 * 2^0]$	7

Table 3.1: The given table shows the conversion of state indices to metastable state.

Alanine tetrapeptide is a three-residue peptide with three *phi* and ψ angles. If we use the first division scheme on all the ϕ's, we will have $2^3 = 8$ metastable states. The naming scheme for these states is as follows:

Similarly, the second division will have a maximum of $4^3 = 64$ states, numbered 0,1,...,63. We will not use the second division here as the first is enough to compare the methods. We know there n several ways to reach these metastable states. This paper aims to develop a method that can sample all the possible metastable states using enhanced sampling. In order to benchmark the proposed mechanism, we can compare it with the following two systems :

a) Simulation using ϕ and ψ - this can act as a reasonable choice of intuitive CV

b) Simulation using TICA collective variable

c) Simulation using SGOOP-d collective variable - this a one-dimensional CV that is known to perform well for this system

We ran five sets of 200ns simulations for all four systems (a, b, c, and proposed mechanism). For the Well-Tempered metadynamics, we used a hill height of 0.4 kJ/mol and a bias factor of 4. Hills were added every 2ps, and a sigma of 0.04 was used.

Chameleon Sequences

Chameleon Sequences[17] (ChSeq) are small peptide sequences that, when present in a protein sequence, can have multiple secondary conformations depending on the interaction in that conformation. The presence of ChSeqs challenges sequence-based secondary structure predictors, and thus these are often used to measure the accuracy of such predictors. The length of ChSeqs is usually in the range of 6-10 residues.

For our studies, we are using a ChSeq with seven residue lengths named Chameleon-HE[18] sequence (RVQDNIV). It can have an alpha helix (1N81) and a beta-sheet (1QDL). In terms of free energy, it can have at least three minima - Alpha helix, Beta Sheet and Random Coil. This is an ideal system to apply our method and explore the conformation. Rather than using the ChSeq alone, we are using a 16-residue sequence extracted from the two proteins - HE-α (residue: 90-105 of PDB 1N81) and HE-β (residue: 224-239 of PDB 1QDL). As the name suggests, HE-α is primarily the alpha helix and HE-β has an anti-parallel beta-sheet structure. The peptide sequences are capped with acetylated N-terminal (ACE) and N-methyl amidated C-terminal (NME) protecting groups. This is done to handle charged end groups, which might prevent the secondary structure formation. We will start with the alpha/beta structure and use our method to explore the free energy space. The structure of the Autoencoder model used here is as follows:

$$64 - 32 - 16 - 8 - 2(Bottleneck) - 2(1Dcircular) - 8 - 16 - 32 - 64$$

Similar to the previous case, the bottleneck layer coordinates are used as CV for the simulations. Unlike the previous case, using ϕ and ψ based division scheme alone is insufficient to study the phase space. Instead, we rely on the Ramachandran plot to distinguish the position of ϕ and ψ. Ramachandran plot is handy in determining the secondary structure of a poly-peptide sequence. We plot Ramachandran plot for a frame of the peptide in time and look into the distribution of residues. They can be an alpha helix, beta-sheet, or random coil. This metric helps us understand the overall secondary structure of the peptide. Since

it is a niche system, we will not be able to benchmark it with other known methods. Thus we will only be comparing it with an intuitive collective variable - average of psi angles. We ran a total of 600ns simulation for both (intuitive and proposed). For the Well-Tempered metadynamics, we used a hill height of 0.4 kJ/mol and a bias factor of 10. Hills were added every 2ps, and a sigma of 0.04 was used.

Chapter 4

Results and Discussion

4.1 Alanine Tetrapeptide

Efficient Sampling of the phase space is a relevant criteria to validate the model. For that we plotted the average of percentage time spend in each metastable state as a bar plot (figure 4.1).

We see that an intuitive collective variable phi-psi can only sample 3 out of 7 metastable states (namely state 0, 2, and 4) correctly, even after 200 ns. These are most likely the major minimal for this system. If we look into other plots, we see that these states are sampled maximally by all the methods. TICA can perform better than phi-psi, but it cannot sample state 7 in all three simulations. SGOOP-d did not perform as well as TICA, as, in one of the simulations, it could not identify state 3. Even though the plot for metastate 3 looks identical for TICA, SGOOP-d and the proposed method, SGOOP-d has a very large error bar indicating only few of the time it could sample this state properly. From this we can say that the proposed method is at least on par with the best of the collective variables.

If we look into the system's three-dimensional free energy plot, the metastable states are arranged as shown in figure 4.2. There are a total of 12 possible direct transitions. After comparing each method's transitions, we see that a simple collective variable like phi-psi can only produce transition in 6 of the possible paths. The TICA and SGOOP-d can cover almost all the paths except the transition between the deepest minima, i.e., minima labeled

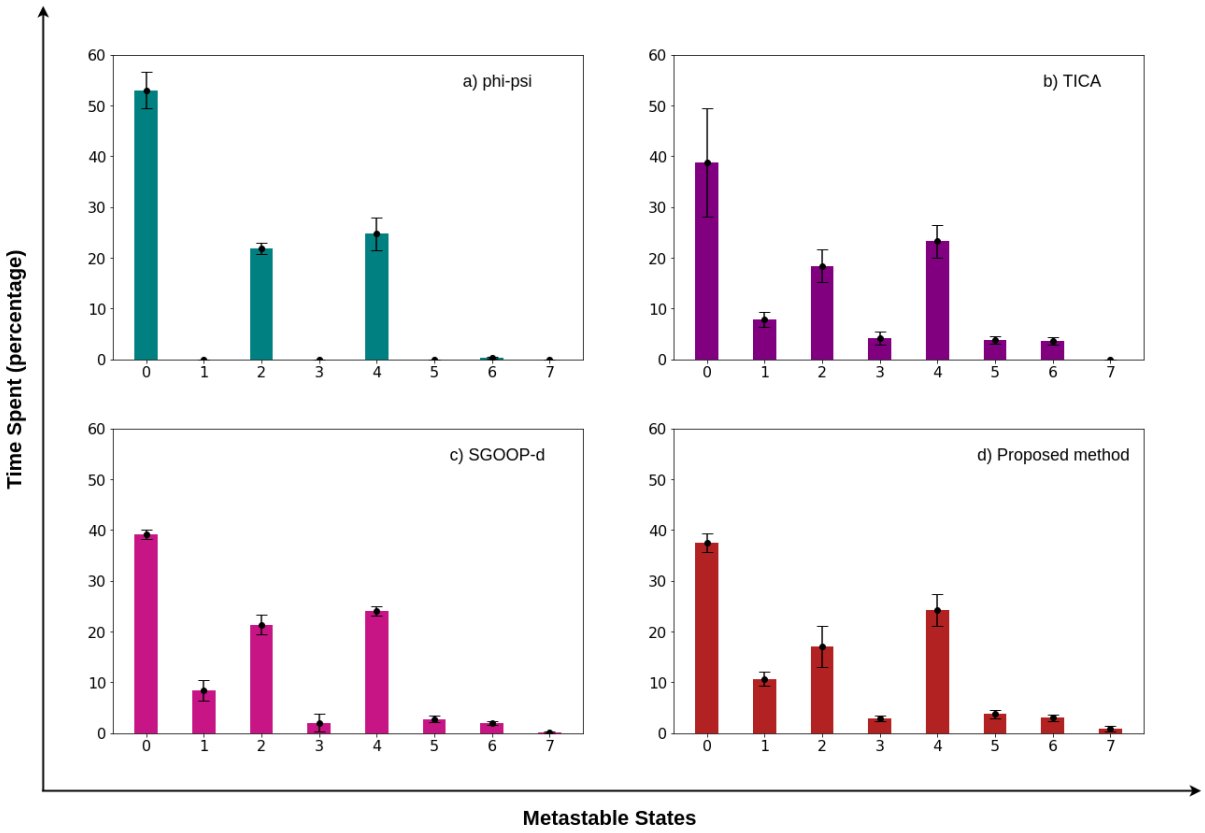


Figure 4.1: The plot shows the average percentage of time spent in each metastable state. a) phi-psi, b) TICA, c) SGOOP-d, d) Proposed method.

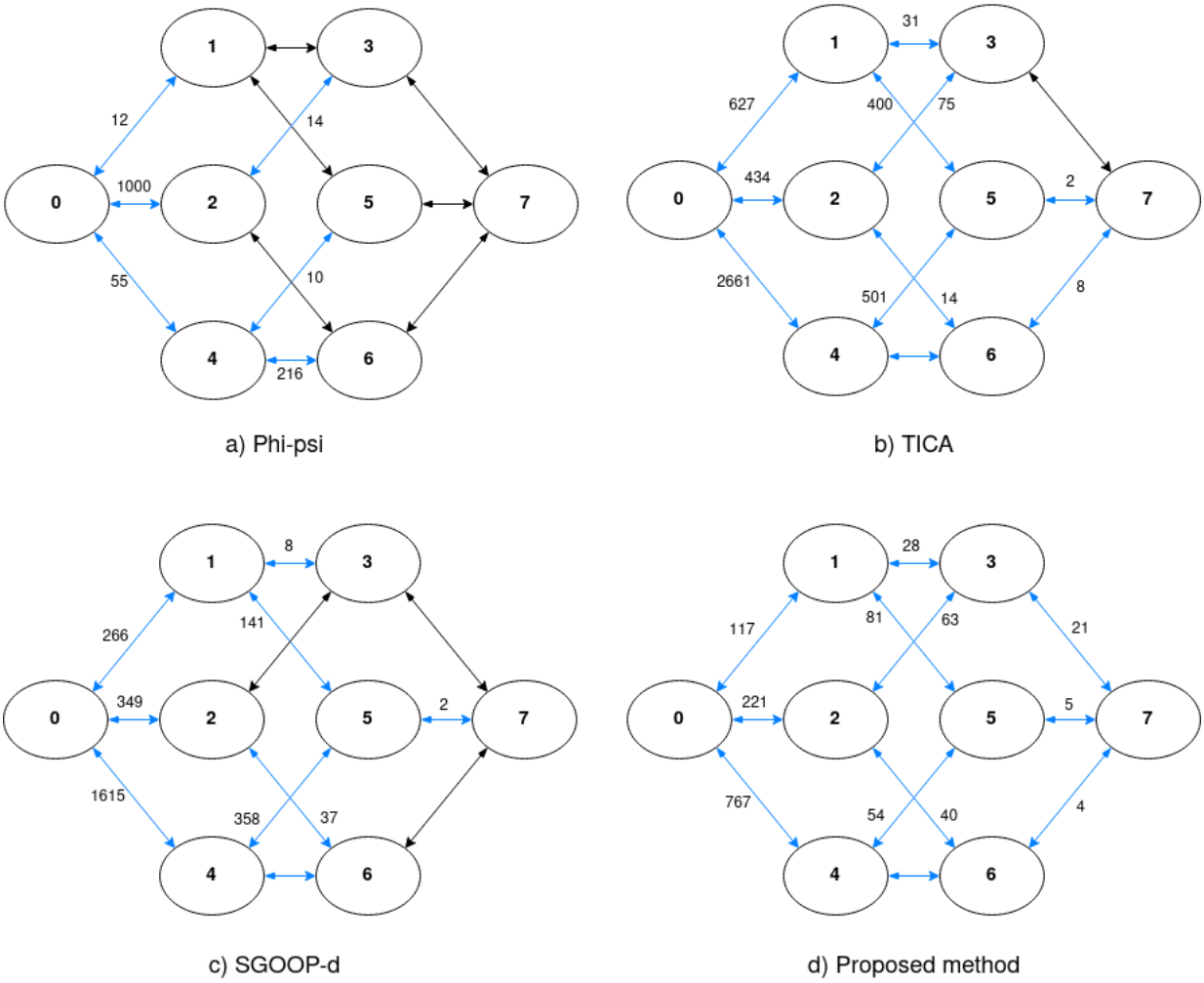


Figure 4.2: The plot shows the transitions between different metastable states along their possible paths in a 3D plane. a) phi-psi, b) TICA, c) SGOOP-d, d) Proposed method.

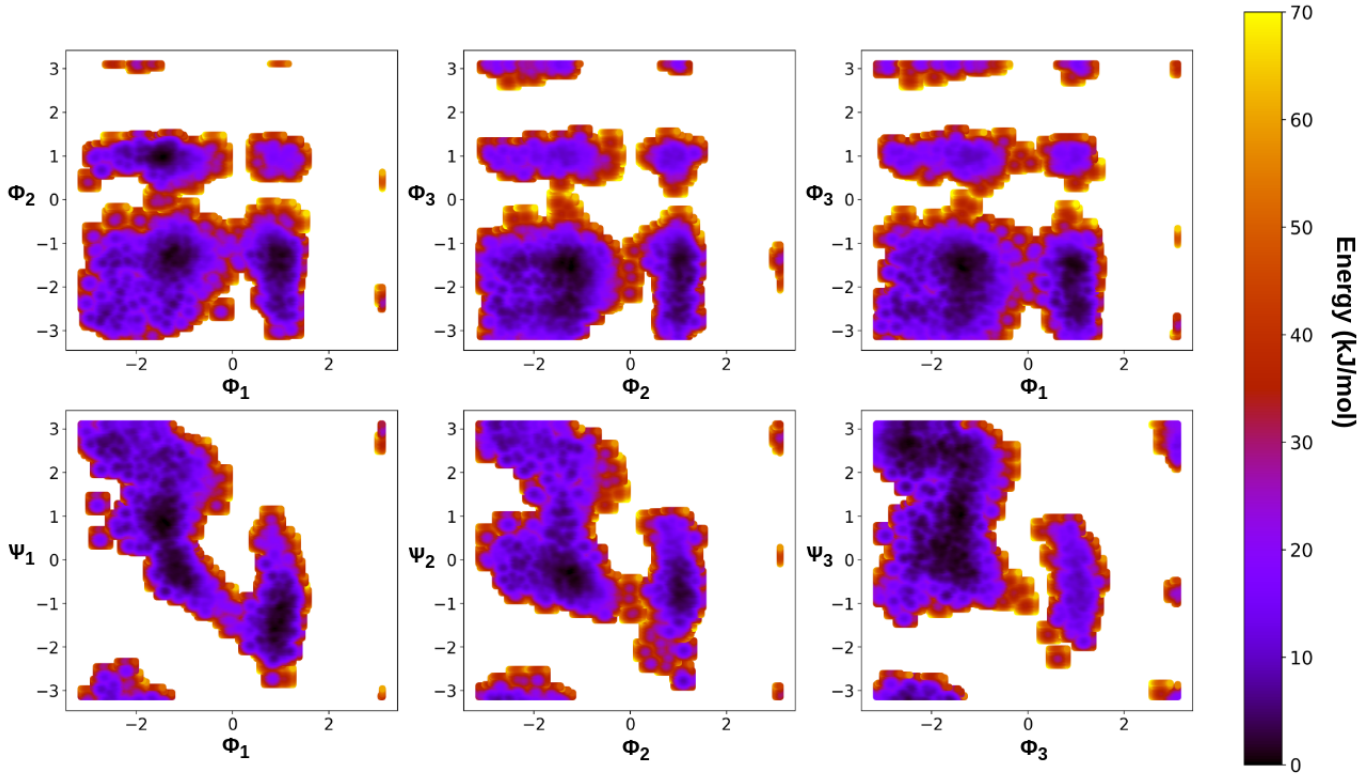


Figure 4.3: The plot shows the two-dimensional free energy surface along all the dihedral angles.

7. We see that all the methods struggled to get transitions from minima 7. TICA and SGOOP-d were unable to sample at least one such possible transition, but the proposed method got a few transitions. Now we see that the proposed method is able to out perform other collective variables.

Looking into the two-dimensional free energy plot (figure 4.3) along all the dihedral angles, we see that this method accurately captures all the significant minimas. The plot can be qualitatively compared with the actual free energy plot.

4.2 Chameleon Sequences

A collective variable should be able to differentiate between the relevant metastable state and promote transitions between them. The metastable states are alpha helix, beta sheet, and

random coil. A Ramachandran plot was drawn for specific frames to visualize the transitions (see Figure 4.5). Starting from residues majorly in the alpha helix, we get to a state where most residues are in the beta-sheet. In order to study the structure at each stage of the simulation, we used DSSP[19], a standard program used for secondary structure prediction. We used the DSSP implementation of gromacs for this calculation. We plotted the number of residues in the alpha helix, beta sheet, and random coil (figure 4.4 shows the same). This plot will show the collective variable's ability to sample the conformations. We used an intuitive collective variable - average of psi for comparison. This is a reasonably good CV as the secondary structures alpha and beta are primarily separated in psi coordinate.

The plot (figure 4.4) shows the different regions A, B, and C. These are regions where one of the structures dominates the others. Looking into the data of the proposed method, we see that it gradually decreases in region A, starting from a 100% alpha structure. Region B shows the time when helical structure dominates with an average of 80% within the region. Region C is where the beta structure dominates with roughly 90%. The intuitive collective variable was able to sample region C, but very few transitions are observed. In both cases, the alpha structure, once collapsed, does not comes up. This could be because the peptide is small, and we are not using any helix-stabilizing agents in the solution. Beta Sheet and Coil are the two primary stable structures for this peptide, as shown by the plot.

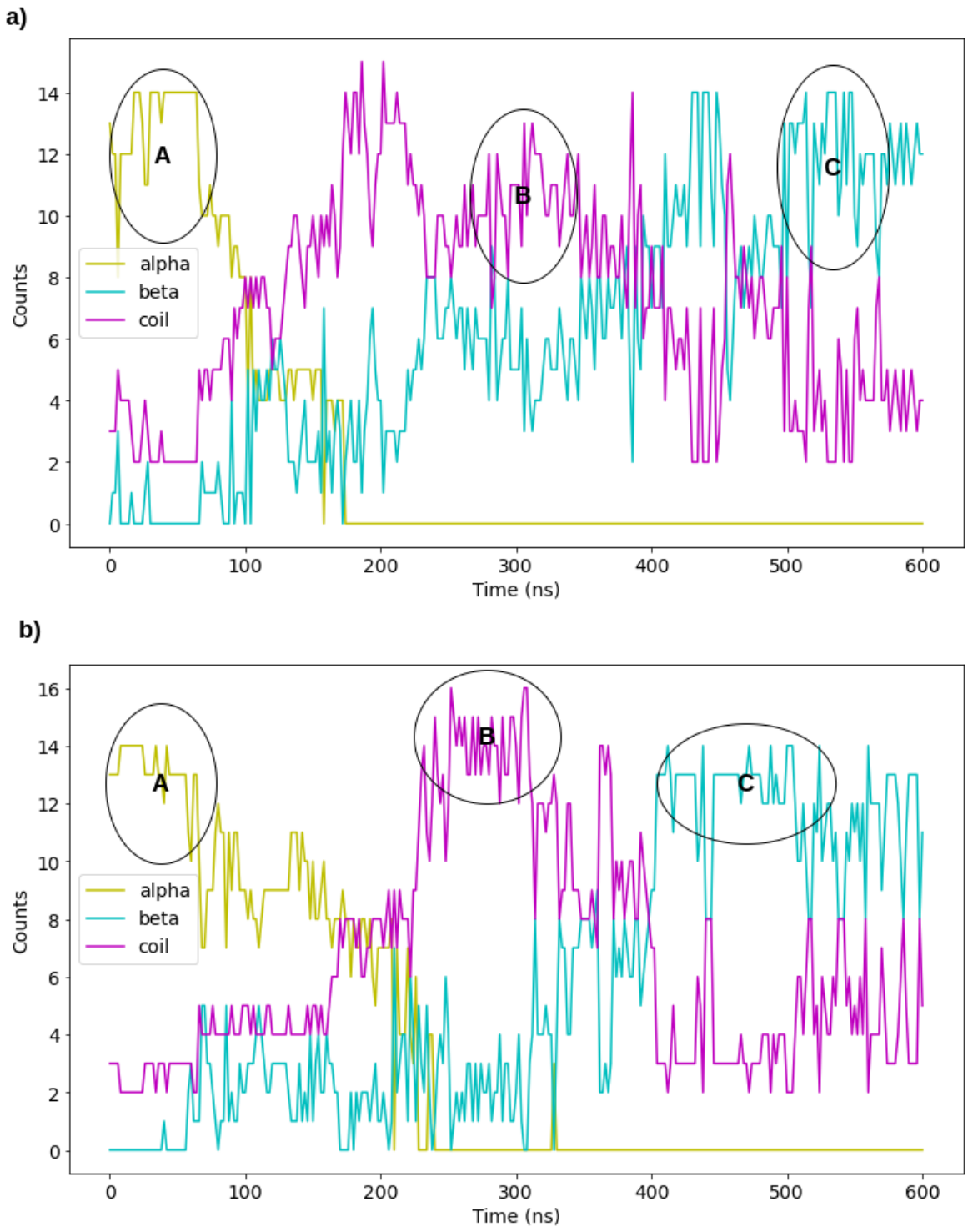


Figure 4.4: The figure shows the number of residues in the particular secondary structures. a) Proposed method, b) Intuitive Collective Variable.

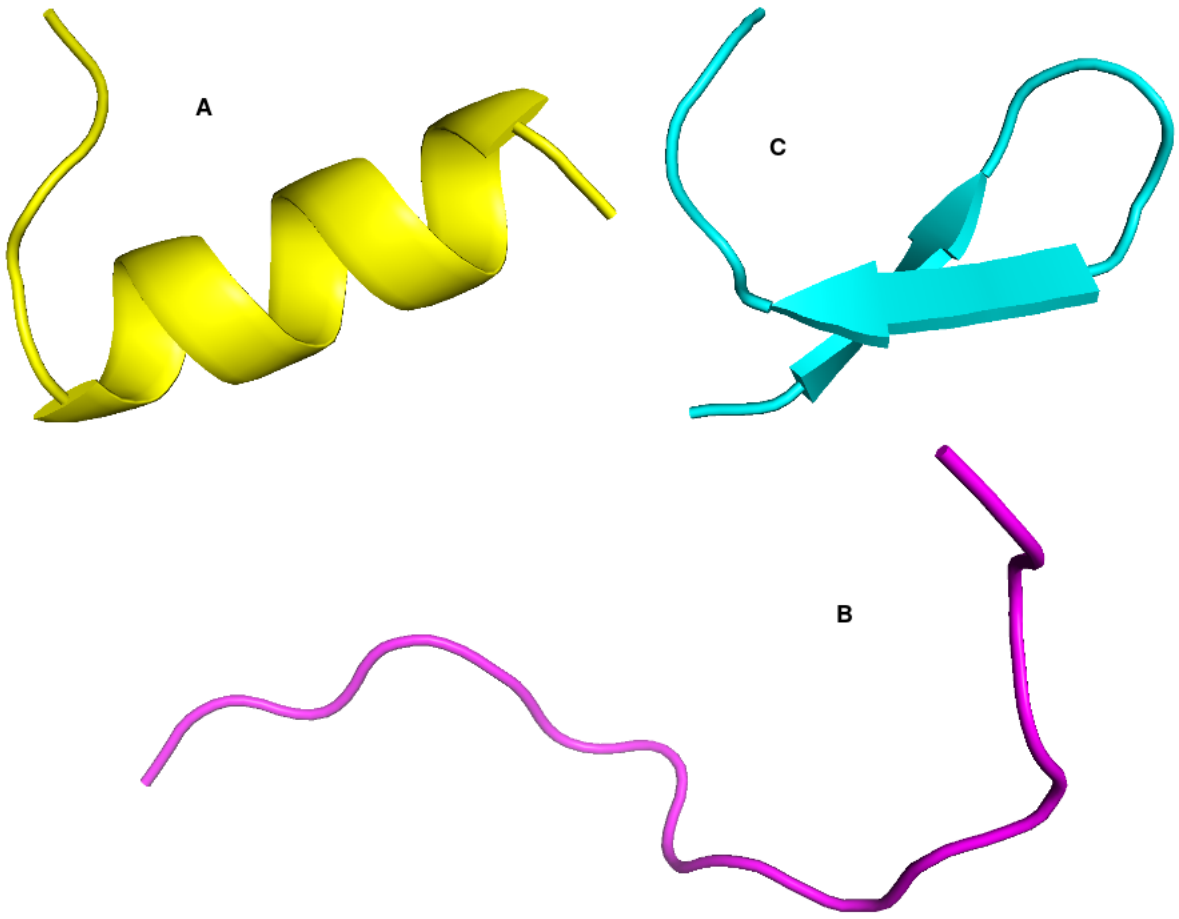


Figure 4.5: The plot shows the various secondary structure possible for the Chameleon sequence. A - helix, B - Coil, and C - Beta-sheet.

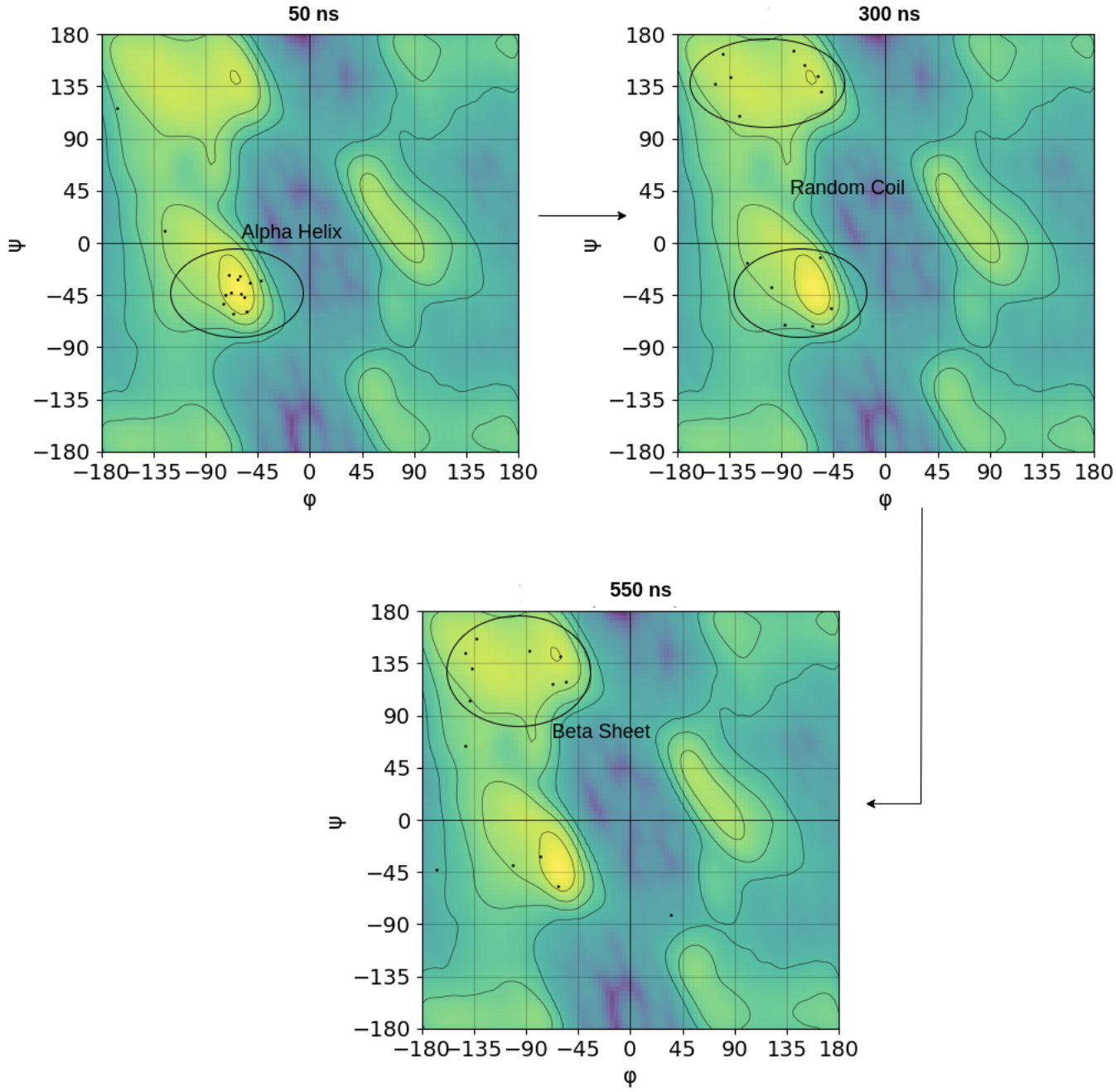


Figure 4.6: The figure shows the Ramachandran plot for the chameleon sequence. We have taken specific frames from the trajectory to visualize the transition from one secondary structure to another.

Chapter 5

Conclusion and Future Outlooks

For Alanine Tetrapeptide, we were able to see that the proposed method performs better and with faster convergence compared to the bench-marked methods. From the free energy plot we also see that the proposed method was able to capture the landscape to a reasonable degree of accuracy. For a larger peptide like Chameleon sequence, the model was able to initiate transitions from one secondary structure to another, but we were not sure how much better is our method compared to a well designed intuitive collective variable. Similar to the Alanine tetrapeptide, we also plotted the free energy. The free energy was plotted along the direction of root mean square deviation (backbone atoms) and radius of gyration (heavy atoms). We see that the proposed method could find slightly more minimas compared to the intuitive collective variable. But we have not done comprehensive study on the structure of the minima found.

Thus we would like to perform extensive study on the free energy surface for this system and understand each of the shown minima. Further, we would also like to extend this method to a very large protein - Human Prion protein and study the misfolding. It is very difficult to get an intuitive collective variable for such a complicated system. Our lab has tried to tackle it using other enhanced sampling techniques. But if this proposed method samples the phase space completely, one could easily learn the mechanism of misfolding.

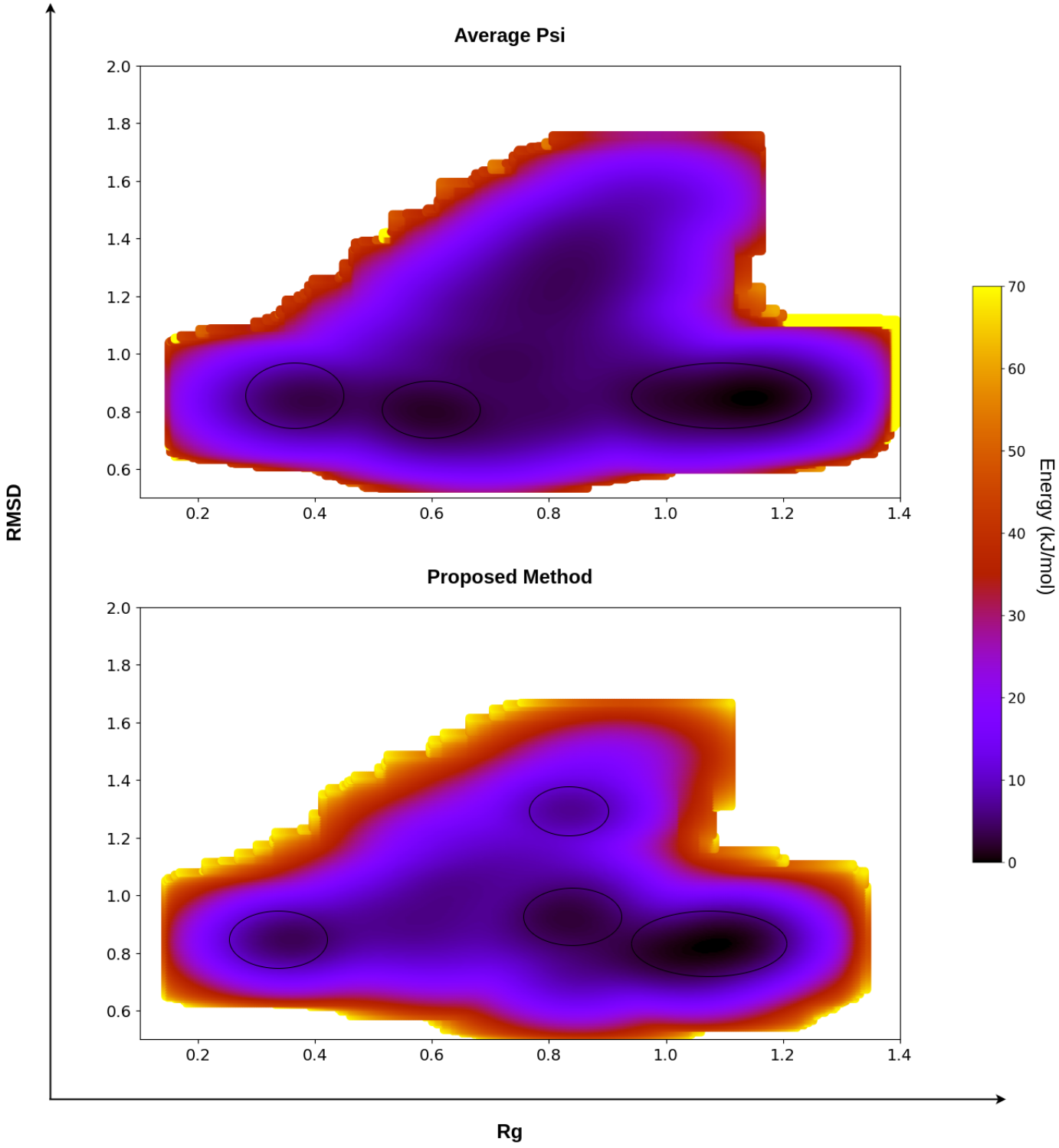


Figure 5.1: The figure shows the 2-dimensional free energy surface along rmsd and rg for the chameleon sequence. a) Average Psi and b) Propsed Method.

Bibliography

[1] Shozeb Haider, Gary N. Parkinson, and Stephen Neidle. Molecular dynamics and principal components analysis of human telomeric quadruplex multimers. *Biophysical Journal*, 95(1):296–311, 2008.

[2] Pratyush Tiwary and B. J. Berne. Spectral gap optimization of order parameters for sampling complex molecular systems. *Proceedings of the National Academy of Sciences*, 113(11):2839–2844, 2016.

[3] Sun-Ting Tsai, Zachary Smith, and Pratyush Tiwary. Sgoop-d: Estimating kinetic distances and reaction coordinate dimensionality for rare event systems from biased/unbiased simulations. *Journal of chemical theory and computation*, 2021.

[4] James McCarty and Michele Parrinello. A variational conformational dynamics approach to the selection of collective variables in metadynamics. *The Journal of Chemical Physics*, 147(20):204109, nov 2017.

[5] Luigi Bonati, GiovanniMaria Piccini, and Michele Parrinello. Deep learning the slow modes for rare events sampling. *Proceedings of the National Academy of Sciences*, 118(44):e2113533118, 2021.

[6] Luigi Bonati, Valerio Rizzi, and Michele Parrinello. Data-driven collective variables for enhanced sampling. *The Journal of Physical Chemistry Letters*, 11:2998–3004, 04 2020.

[7] Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the National Academy of Sciences*, 99(20):12562–12566, 2002.

[8] Alessandro Barducci, Giovanni Bussi, and Michele Parrinello. Well-tempered metadynamics: A smoothly converging and tunable free-energy method. *Phys. Rev. Lett.*, 100:020603, Jan 2008.

[9] C Ramakrishnan and G N Ramachandran. Stereochemical criteria for polypeptide and protein chain conformations. II. allowed conformations for a pair of peptide units. *Biophys J*, 5(6):909–933, November 1965.

[10] Dor Bank, Noam Koenigstein, and Raja Giryes. Autoencoders. *CoRR*, abs/2003.05991, 2020.

[11] Peter J. Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[12] Hythem Sidky, Wei Chen, and Andrew L. Ferguson. Machine learning for collective variable discovery and enhanced sampling in biomolecular simulation. *Molecular Physics*, 118(5):e1737742, 2020.

[13] H BEKKER, HJC BERENDSEN, EJ DIJKSTRA, S ACHTEROP, R VONDRUMEN, D VANDERSPOEL, A SIJBERS, H Keegstra, and MKR RENARDUS. Gromacs - a parallel computer for molecular-dynamics simulations. In RA DeGroot and J Nadrchal, editors, *PHYSICS COMPUTING '92*, pages 252–256. World Scientific Publishing, 1993. 4th International Conference on Computational Physics (PC 92) ; Conference date: 24-08-1992 Through 28-08-1992.

[14] Lindahl, Abraham, Hess, and van der Spoel. Gromacs 2021.4 manual, November 2021.

[15] Gareth A. Tribello, Massimiliano Bonomi, Davide Branduardi, Carlo Camilloni, and Giovanni Bussi. PLUMED 2: New feathers for an old bird. *Computer Physics Communications*, 185(2):604–613, feb 2014.

[16] Kresten Lindorff-Larsen, Stefano Piana, Kim Palmo, Paul Maragakis, John L Klepeis, Ron O Dror, and David E Shaw. Improved side-chain torsion potentials for the amber ff99SB protein force field. *Proteins*, 78(8):1950–1958, June 2010.

[17] Mihaly Mezei. Revisiting chameleon sequences in the protein data bank. *Algorithms*, 11(8), 2018.

[18] Bongkeun Kim, Thanh D. Do, Eric Y. Hayden, David B. Teplow, Michael T. Bowers, and Joan-Emma Shea. Aggregation of chameleon peptides: Implications of -helicity in fibril formation. *The Journal of Physical Chemistry B*, 120(26):5874–5883, 2016. PMID: 27001160.

[19] Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637, 1983.