Development And Application Of Machine Learning Approaches For Prediction, Identification And Sampling Problems In the Field Of Molecular Modeling and Simulation

Thesis Submitted to the

University of Calcutta

For the Degree of Doctor of Philosophy (Science) by

Dibyendu Maity

Department of Chemical and Biological Sciences

S.N. Bose National Centre for Basic Sciences

Kolkata-700106

July 19, 2026

DECLARATION

I hereby declare that the research work presented in this thesis entitled **"Development And Application Of Machine Learning Approaches For Prediction, Identification And Sampling Problems In the Field Of Molecular Modeling and Simulation"** has been carried out by me at the Department of Chemical and Biological Sciences, S.N. Bose National Centre for Basic Sciences, Kolkata-700106, and it has not been submitted elsewhere for any degree or diploma.

Date: July 19, 2026

Dibyendu Maity

Dedicated to Mahākāla, the eternal and boundless Time that remains untouched by you and me.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor, **Prof. Suman Chakrabarty**, for his guidance, encouragement, and support throughout this work. His advice and scientific insight have been invaluable during the development of this thesis.

I am also thankful to all my group members for their help, discussions, and cooperation. Their support made the research environment productive and enjoyable, and their suggestions helped me improve different parts of this work.

Finally, I am deeply grateful to my family for their constant love, patience, and encouragement. Their support has been a source of strength throughout my research journey.

Abstract

Molecular motion is central to many processes in chemistry, biology, and materials science. A molecule dissolves in water, a liquid turns into ice, a drug leaves a protein binding site, or a protein chain folds into a stable shape because atoms move, interact, and rearrange over time. Computer simulations can follow these motions in great detail, but two practical problems appear again and again. First, the amount of molecular information is very large. Even a small system contains many atoms, and each atom has a position and a velocity that change with time. Second, the most important events are often rare. A simulation may run for a long time while the system only vibrates near its current state, even though the event of interest is a transition to a different state.

This thesis develops machine-learning-assisted methods to make molecular simulation more useful for prediction, identification, and sampling. In this work, machine learning is not used as a replacement for physical models. Instead, it is used as a practical tool for organizing molecular information, finding useful patterns, choosing where new simulations should start, and checking whether a model continues to work when it is moved outside the data on which it was trained. The central idea is simple: a model is useful only if it keeps contact with molecular physics. A prediction should work on unfamiliar chemistry, a structural map should correspond to real molecular environments, and a sampling method should produce pathways that can be followed and interpreted.

The first part of the thesis studies how molecules should be represented when predicting aqueous solvation free energy. This property measures how favorable it is for a molecule to move from the gas phase into water. It is closely related to solubility, partitioning, and molecular recognition, so it is an important test case for molecular prediction. Several ways of describing a molecule are compared, including simple physical descriptors, structural fingerprints, molecular graphs, and a representation learned from molecular trajectories. The main question is not only which model gives the smallest error on a random test set, but which representation remains reliable when the test molecules come from a different chemical distribution.

The results show that good performance on familiar data can be misleading. Simple descriptors such as polar surface area and hydrogen-bond counts capture important trends, but they may miss where polar groups are located and whether they are exposed to water. Fingerprints recognize local fragments, but they do not always capture how solvation changes with molecular size and shape. Graph models include atom-level connectivity and can describe solute-solvent interactions more directly, yet they can still fail for chemical classes that are poorly represented in training. Long-chain alkanes,

shielded polar aromatic molecules, ionizable neutral molecules, and rare elements expose these limitations. The chapter therefore argues that transferability must be tested directly. A model should not be called reliable only because it performs well on a random split of one dataset.

The trajectory-based representation studied in this thesis improves part of this picture. By using information from molecular conformations and simple trajectory-level physical quantities, it gives the prediction model access to shape and flexibility information that is missing from many two-dimensional descriptions. The best results are obtained when this learned information is combined with clear indicators of the model's range of support, such as flags for unusual chemistry, shielded polarity, fused aromatic systems, and elements not well represented in training. This does not remove every error, but it makes the failures easier to understand. The lesson is that representation learning and domain awareness must work together.

The second part of the thesis moves from predicting a property of whole molecules to identifying local structures in water and ice. Water has many solid forms, and the local arrangement around each molecule can change under temperature, pressure, and coexistence with liquid water. Traditional structural measures are useful, but they can overlap when several ice phases and liquid-like environments are present together. This makes automatic phase identification difficult, especially in simulations where boundaries and defects are important.

To address this problem, the thesis introduces IceCoder, a method that learns a compact map of local water environments. Each local environment is described in a way that respects the surrounding molecular geometry, and the learned map places similar environments near one another. In this map, liquid water and several ice forms separate into meaningful regions. The method is tested not only by checking whether known structures are classified correctly, but also by following simulations in which phases grow, melt, or coexist. This is important because a useful structural map should not simply label ideal crystal structures. It should also track the gradual and imperfect changes that happen in real simulations.

The third part of the thesis focuses on rare molecular transitions. Many important processes, such as ligand unbinding or protein folding, happen through more than one possible route. A single distance or a single progress variable may hide this diversity. The thesis develops and applies a pathway-based workflow that first generates possible transition routes and then measures how much each route contributes to the overall process. The workflow uses short simulation segments to build candidate pathways, groups similar pathways together, refines them into smooth reference routes, and then uses many parallel simulations to estimate route-specific behavior.

This approach is applied to model systems, ligand unbinding, and protein folding. In these examples, the method shows that different pathways can carry different kinetic

importance. For example, changes in a protein can redirect how a drug leaves its binding site, and different unbinding routes can have different rates. The purpose is not only to speed up simulation, but also to turn a rare event into a set of interpretable molecular stories. Instead of asking only how long a transition takes on average, the workflow asks which routes are used, how often they are used, and what structural changes define them.

The final part of the thesis develops TRAILS-MD, an adaptive sampling framework for exploring molecular motion when the important pathway is not known in advance. The method runs many short simulations, learns from the structures already visited, and starts new simulations from regions that appear sparse, unusual, or promising for further exploration. A key feature is lineage tracking: every new simulation segment records which earlier segment produced it. This record makes it possible to reconstruct continuous histories from many short pieces. Without this information, an adaptive simulation may appear to cover a broad region of molecular space, but it may be unclear whether the sampled points are connected by physically meaningful motion.

TRAILS-MD is tested on folding and unbinding problems where broad exploration and mechanistic interpretation are both needed. The results show that adaptive sampling becomes more useful when exploration is tied to history. The method can locate new regions, improve coverage of important molecular states, and provide starting points for more detailed kinetic calculations. At the same time, lineage information helps distinguish real pathways from accidental overlap in a low-dimensional plot.

Across all chapters, the thesis develops one connected view of machine learning in molecular simulation. Machine learning is valuable when it helps decide what information should be kept, how molecular states should be organized, where simulation effort should be spent, and how results should be checked. It is less useful when it only improves a score on data that are too similar to the training set or when it produces a map that cannot be interpreted physically. For this reason, the thesis repeatedly uses external tests, structural validation, pathway reconstruction, and kinetic checks. These tests expose where a method succeeds and where it fails.

The overall contribution of this thesis is a set of practical workflows for molecular modeling and simulation. The first workflow tests whether molecular representations can predict solvation free energy beyond familiar chemistry. The second identifies local water and ice environments from simulation data. The third discovers rare-event pathways and connects them to route-specific kinetics. The fourth automates conformational exploration while preserving the history needed to interpret the result. Together, these studies show that the best role for machine learning in molecular simulation is not to replace physical reasoning, but to make it easier to use: by compressing complex data, highlighting hidden structure, directing simulation effort, and making the limits of each model visible.

List of Publications

Publications Included in This Thesis

1. Maity, D.; Chakrabarty, S. IceCoder: Identification of Ice Phases in Molecular Simulation Using Variational Autoencoder. *Journal of Chemical Theory and Computation* **2025**, *21* (4), 1916–1928. https://doi.org/10.1021/acs.jctc.4c01298.

2. Maity, D.; Shahid, S.; Chakrabarty, S. PathGennie: Rapid Generation of Rare Event Pathways via Direction-Guided Adaptive Sampling Using Ultrashort Monitored Trajectories. *Journal of Chemical Theory and Computation* **2025**, *21* (22), 11377–11389. https://doi.org/10.1021/acs.jctc.5c01244.

3. Maity, D.; Chakrabarty, S. Challenges in Transferable Prediction of Solvation Free Energy: A Comparative Analysis of Molecular Representations and Machine Learning Methods. *Research Square* **2025**. https://doi.org/10.21203/rs.3.rs-6727155/v1.

4. Maity, D.; Shahid, S.; Bhattacharya, S.; Majumdar, R.; Chakrabarty, S. Quantitative Pathway-Resolved Kinetics from Neural Network-Guided Weighted Ensemble Simulations. *ChemRxiv* **2026**. https://doi.org/10.26434/chemrxiv.15005553/v1.

5. Maity, D.; Majumdar, R.; Chakrabarty, S. TRAILS-MD: Mapping Complex Conformational Landscapes and Transition Pathways via Lightweight Lineage-Aware Adaptive Sampling. Submitted.

Publications Not Included in This Thesis

1. Sahoo, R.; Maity, D.; Shankar Rao, D. S.; Chakrabarty, S.; Yelamaggad, C. V.; Prasad, S. K. Dimer-Parity-Dependent Odd–Even Effects in Photoinduced Transitions to Cholesteric and Twist Grain Boundary Smectic-C* Mesophases: Experiments and Simulations. *Physical Review E* **2022**, *106* (4), 044702. https://doi.org/10.1103/PhysRevE.106.044702.

2. Bhaumik, S. K.; Maity, D.; Basu, I.; Chakrabarty, S.; Banerjee, S. Efficient Light Harvesting in Self-Assembled Organic Luminescent Nanotubes. *Chemical Science* **2023**, *14* (16), 4363–4374. https://doi.org/10.1039/D3SC00375B.

3. Shahid, S.; Maity, D.; Chakrabarty, S. CoWERA: A Temporal Coherence Guided Binless Resampling Algorithm for Weighted-Ensemble Based Estimation of Rare-Event Kinetics. *The Journal of Chemical Physics* **2026**, *164* (13), 134111. https://doi.org/10.1063/5.0320586.

4. Pal, S.; Maity, D.; Chakraborty, J.; Jha, S.; Sarma, K.; Koner, N.; Chakrabarty, S.; Das, D. Pathway Controlled Phase Separation of Minimal Building Blocks Utilizing a Dissociative Chemical Transformation. *Angewandte Chemie International Edition* **2026**, e1914460. https://doi.org/10.1002/anie.1914460.

5. Sarkar, S.; Saha, R.; Dutta, A.; Chattopadhyay, R.; Maity, D.; Biswas, I.; Mondal, A.; Ghosh, A.; Ganguly, R.; Santra, A. K.; Garain, U.; Mitra, D.; Chakrabarty, S. Microwave-Assisted Thermal Profiling of Blood: A Potential Biomarker for Differentiating Cancer and Non-Cancer States. *Journal of Medical Engineering & Technology* **2026**, 1–16. https://doi.org/10.1080/03091902.2026.2698512.

Contents

List of Publications i

1 Introduction 1
- 1.1 Molecular Modeling Across Scales . . . 2
- 1.2 Statistical Mechanics and Molecular Dynamics . . . 3
- 1.3 Molecular Models and the Resolution–Efficiency Tradeoff . . . 3
- 1.4 The Timescale and Rare-Event Problem . . . 6
- 1.5 The Dimensionality and Representation Problem . . . 7
- 1.6 Collective Variables and the Identification Bottleneck . . . 10
- 1.7 Enhanced Sampling Methods . . . 11
  - 1.7.1 Collective-Variable Biasing . . . 11
  - 1.7.2 Generalized-Ensemble Methods . . . 13
  - 1.7.3 Path-Based, Event-Based, and Weighted-Ensemble Methods . . . 13
  - 1.7.4 Adaptive and ML-Guided Sampling . . . 14
- 1.8 Machine Learning in Molecular Modeling and Simulation . . . 16
  - 1.8.1 ML for Molecular Property Prediction . . . 16
  - 1.8.2 ML for State and Phase Identification . . . 16
  - 1.8.3 ML for Collective-Variable Discovery . . . 17
  - 1.8.4 ML-Guided Adaptive Sampling . . . 17
  - 1.8.5 ML Interatomic Potentials and Foundation Models . . . 18
- 1.9 Toward Automated Molecular Discovery Workflows . . . 18
- 1.10 Thesis Scope and Chapter Roadmap . . . 19
- 1.11 Objectives and Contributions . . . 20

2 Theoretical Background and Methodology 22
- 2.1 Molecular Dynamics Foundations . . . 22
  - 2.1.1 Verlet and Velocity Verlet Integrators . . . 23
  - 2.1.2 Leapfrog Integrator . . . 24
  - 2.1.3 Langevin Dynamics . . . 24
- 2.2 Potential Energy Surfaces and Force Fields . . . 25
  - 2.2.1 The Born–Oppenheimer Approximation . . . 25
  - 2.2.2 Functional Form of Empirical Force Fields . . . 25
- 2.3 Molecular Representations . . . 28
  - 2.3.1 Coordinate Invariances . . . 28

2.3.2 Descriptor-Based Molecular Representations 29
2.3.3 Molecular Fingerprints 29
2.3.4 Graph-Based Molecular Representations 29
2.3.5 Collective Variables in Molecular Dynamics 30
2.3.6 Classical Order Parameters for Water and Ice 31
2.3.7 Smooth Overlap of Atomic Positions (SOAP) 33
2.4 Dimensionality Reduction and Latent Variables 34
2.4.1 Principal Component Analysis 34
2.4.2 Autoencoders 35
2.4.3 Variational Autoencoders 35
2.4.4 Time-Lagged Projection Models 36
2.5 Supervised Learning for Prediction and Classification 37
2.5.1 Gradient-Boosted Decision Trees: XGBoost 38
2.5.2 Random Forest Regressor 38
2.5.3 Kernel Ridge Regression 38
2.5.4 Support Vector Regression and Support Vector Classification . . 39
2.5.5 Chemically Interpretable Graph Interaction Network 40
2.6 Rare-Event and Path-Based Methods 40
2.6.1 Weighted Ensemble Simulations 41
2.6.2 Path Collective Variables 45
2.6.3 Dynamic Time Warping for Transition Path Analysis 45
2.7 Markov State Modeling and Kinetic Analysis 46
2.7.1 State Discretization 46
2.7.2 Transition Count and Transition Probability Matrices 47
2.7.3 Markovianity and Implied Timescales 47
2.7.4 Chapman–Kolmogorov Validation 49
2.7.5 Kinetic Observables 49
2.7.6 Coarse-Graining and Interpretation 50
2.7.7 Relation to Variational Kinetic Models 50

3 Transferability of Molecular Representations for Aqueous Solvation Free-Energy Prediction 51
3.1 Motivation and Physical Context 51
3.2 Datasets and Curation with Leakage Awareness 53
3.3 A Progression of Molecular Representations 54
3.3.1 Descriptor Representation 54
3.3.2 Fingerprint Representation 55
3.3.3 Graph Representation 55
3.3.4 Trajectory-Derived Molecular Representation 55

3.4 Chemical-Space Structure and Representation Maps . . . 57
3.5 Prediction Models and Validation Protocol . . . 58
3.6 Results: What Each Representation Learns and Misses . . . 60
3.6.1 Global Descriptors Capture Dominant Trends but Miss Chemical Context . . . 62
3.6.2 Fingerprints Resolve Local Motifs but Not Thermodynamic Scaling 63
3.6.3 Graph Models Add Atomistic Detail but Do Not Guarantee Extrapolation . . . 65
3.6.4 Trajectory-Derived Features Improve Transfer but Do Not Remove Chemotype-Specific Failures . . . 65
3.7 Conclusion . . . 69

4 IceCoder: Self-Supervised Identification of Ice Polymorphs 71
4.1 Introduction . . . 71
4.2 Motivation . . . 71
4.2.1 Selection of Baseline Order Parameters . . . 73
4.2.2 Baseline Performance and Limitations . . . 74
4.3 IceCoder Framework . . . 74
4.3.1 SOAP Representation . . . 74
4.4 Variational Autoencoder . . . 75
4.5 Simulation Data . . . 76
4.6 Results and Discussion . . . 77
4.6.1 Separation of Ice Phases in Latent Space . . . 77
4.6.2 Intra-Phase Resolution . . . 78
4.6.3 Automated Phase Assignment . . . 79
4.6.4 Two-Phase Ice Growth . . . 80
4.6.5 Three-Phase Conversion . . . 80
4.7 Conclusions . . . 81

5 PathGennie and Path-Resolved Weighted Ensemble Kinetics: From Rapid Pathway Discovery to Quantitative Route-Specific Rate Constants 84
5.1 Introduction . . . 84
5.2 Background and Motivation . . . 86
5.3 The PathGennie Algorithm . . . 87
5.4 Simulation Systems and Computational Details . . . 89
5.4.1 Force Fields and System Preparation . . . 89
5.4.2 Equilibration Protocol . . . 90
5.4.3 Collective Variable Construction for Ligand Unbinding . . . 90
5.5 Robustness to an Imperfect Progress Variable . . . 90

5.6 Ligand Unbinding Pathways . . . 92
5.6.1 Benzene Dissociation from T4 Lysozyme L99A . . . 92
5.6.2 Imatinib Unbinding from Abl Kinase . . . 93
5.7 Folding and Unfolding of Fast-Folding Proteins . . . 94
5.8 The Path-Resolved Kinetics Workflow . . . 94
5.8.1 Dynamic Time Warping for Path Classification . . . 96
5.8.2 Neural Network Path Refinement and PathCV Construction . . . 97
5.8.3 Iterative Path Refinement and Convergence . . . 98
5.8.4 Pre-Seeded WE Sampling Along Path CVs . . . 99
5.9 Path-Informed Weighted Ensemble Simulations . . . 101
5.10 Applications of Path-Resolved WE . . . 102
5.10.1 Temperature Crossover in a Three-Hole Potential . . . 102
5.10.2 Mutation-Dependent Imatinib Dissociation from Abl Kinase . . . 103
5.10.3 Path-Resolved Dissociation Kinetics of Benzamidine from Trypsin 106
5.11 Discussion . . . 108
5.12 Conclusions . . . 109

6 TRAILS-MD: Lineage-Aware Adaptive Sampling for the Mapping of Complex Conformational Landscapes and Transition Pathways 111
6.1 Introduction . . . 111
6.2 The TRAILS-MD Framework . . . 114
6.2.1 Overall Algorithm . . . 114
6.2.2 Trajectory Propagation . . . 116
6.2.3 Featurization and Coordinate Representation . . . 117
6.2.4 Sampling-Space Projection Models . . . 117
6.2.5 Configurational Space Partitioning and Spawning . . . 119
6.2.6 Sampling Coverage and Convergence . . . 121
6.2.7 Post-Processing and Seeding for Kinetic Analysis . . . 122
6.3 Computational Details . . . 122
6.3.1 Alanine Dipeptide in Explicit Water . . . 123
6.3.2 Chignolin (CLN025) . . . 123
6.3.3 AIB9 Peptide . . . 123
6.3.4 Trp-Cage and WW Domain Protein Folding . . . 124
6.3.5 Ligand Unbinding: OAMe-G2 and HSP90 . . . 124
6.4 Results . . . 125
6.4.1 Alanine Dipeptide: Validating the Framework with a Known Coordinate . . . 125
6.4.2 Chignolin Folding: Comparing Fixed and Learned Coordinate Spaces . . . 126

6.4.3 AIB9 Peptide: Basin Discovery Does Not Imply a Pathway 128
6.4.4 Entropic Bottlenecks in Protein Folding: Trp-Cage and WW Domain 130
6.4.5 Rapid Mapping of Ligand Unbinding Tunnels 131
6.4.6 Markov State Model Construction from TRAILS-MD-Seeded Trajectories 132
6.5 Software, Performance, and Extensibility 135
6.5.1 Computational Performance and Multi-GPU Scaling 135
6.5.2 Software Implementation and Dependencies 136
6.5.3 Extensibility 137
6.6 Discussion 137
6.7 Summary and Outlook 139

7 Summary and Future Outlook **142**
7.1 General Summary 142
7.2 Limitations and Open Questions 144
7.3 Future Research Directions 145
7.3.1 Richer and More Transferable Molecular Representations 145
7.3.2 Adaptive Sampling with Uncertainty and Feedback 145
7.3.3 Pathway-Resolved and Mechanism-Aware Kinetics 146
7.3.4 Toward Autonomous Molecular Discovery 146
7.4 Concluding Perspective 146

List of Figures

1.1 Resolution hierarchy of molecular modeling approaches. Quantum mechanical descriptions treat electrons explicitly and can resolve chemical rearrangement; all-atom classical MD replaces electrons with empirical interaction functions; coarse-grained models map groups of atoms to effective particles; and continuum descriptions replace molecular detail with field-level variables. The same ordering underlies the later choice of compact descriptors, local structural representations, and path-based sampling coordinates. 5

1.2 The timescale gap in molecular simulation. Femtosecond timesteps are required to integrate fast bonded vibrations, while relevant transitions such as ligand binding, conformational switching, nucleation, and folding may occur on microsecond-to-second scales. Enhanced sampling and adaptive simulation strategies are responses to this separation. 7

1.3 Spectrum of molecular representations used in molecular modeling and machine learning. Raw coordinates preserve direct geometry but are high-dimensional; descriptors and fingerprints provide compact hand-crafted encodings; molecular graphs preserve chemical topology; trajectory-derived embeddings summarize conformational and physical information from molecular simulations; local descriptors such as SOAP encode symmetry-aware environments; and learned latent spaces compress molecular data into variables useful for classification, visualization, and sampling. 9

1.4 Classification of enhanced sampling methods. CV-biasing methods alter the effective potential along selected coordinates; generalized-ensemble methods improve exploration through thermodynamic or Hamiltonian exchange; path- and event-based methods sample transition trajectories or interfaces; Weighted Ensemble uses statistically weighted splitting and merging of trajectory ensembles; and adaptive or ML-guided methods update sampling decisions using learned or evolving representations. . 12

2.1 Schematic overview of molecular representations. From SMILES strings, descriptor-based and fingerprint-based representations are generated for classical machine learning models. For graph-based models, molecular graphs are constructed with node features and adjacency matrices and fed to different architectures for mapping to the target property. 30
2.2 Free energy projection of the alanine dipeptide system in low-dimensional CV space. (a) Alanine dipeptide is shown in ball-and-stick model. (b) Free energy surface projected onto the backbone dihedral angles ϕ and ψ. (c) A learned representation of the system in the latent space of a variational autoencoder. 31
2.3 Comparative schematic of descriptor compression using PCA, autoencoders, and variational autoencoders. 35
2.4 Schematic overview of the VAE workflow. The system is featurized using a suitable invariant representation and passed to the encoder network to produce a low-dimensional latent coordinate. The decoder reconstructs the original input from the latent coordinate, and the model is trained by minimizing the negative evidence lower bound. The figure also illustrates a representative training-loss curve and the projection of data into the learned latent space. 36
2.5 Comparison of five supervised machine-learning approaches for molecular property prediction and classification. From left to right, the figure illustrates gradient-boosted trees (XGBoost), random forest regression, kernel ridge regression (KRR), support vector regression/classification (SVR/SVC), and the Chemically Interpretable Graph Interaction Network (CIGIN). Although all methods learn the same mapping from molecular descriptors to target properties, they differ in their underlying learning strategies, ranging from ensemble tree-based methods and kernel-based similarity models to margin-based learning and graph neural networks that explicitly model solute–solvent atom-pair interactions. The schematic highlights the progression from conventional feature-vector methods to graph-native representations capable of learning chemically meaningful intermolecular interactions. . 39
2.6 Schematic of Weighted Ensemble sampling. Walkers are propagated under unbiased dynamics and periodically resampled within bins along a progress coordinate. Splitting increases resolution in under-sampled regions, merging controls computational cost, and statistical weights preserve the correct ensemble. In steady-state calculations, probability flux into a target state yields kinetic information. 43

2.7 Workflow for constructing and analyzing a Markov state model (MSM) from molecular dynamics trajectories. Raw trajectory coordinates (R_t) are projected into a lower-dimensional feature space ((x_t) or (z_t)) and discretized into microstates (s_t). Transition counts ($C_{ij}(\tau)$) are accumulated at lag time (τ) and row-normalized to obtain the transition probability matrix ($T_{ij}(\tau)$). Markovianity is assessed through implied-timescale convergence, after which the kinetic model is coarse-grained into macrostates and represented as a state network. The resulting MSM provides stationary populations (π), transition rates, mean first-passage times (MFPTs), and committor probabilities (q^+) for quantitative analysis of metastable-state interconversion. 48

3.1 Overview of the solvation free-energy datasets used in this chapter. (A) Overlap of chemical entities among MNSol, FreeSolv, and CombiSolv. (B) Functional-group populations in each dataset. (C,D) Two-dimensional t-SNE projections of descriptor and PubChem fingerprint representations. (E) Two-dimensional latent embedding learned from molecular graphs using a GCN-VAE model. 53

3.2 Trajectory-based molecular representation-learning workflow. Molecular dynamics frames are encoded using a coordinate-equivariant graph neural network pretrained against energetic components, global shape and physical observables, and a coordinate-denoising objective. Frame-level embeddings are combined through attention pooling, while predicted trajectory observables are summarized by their means and standard deviations. The final 148-dimensional representation combines a 128-dimensional pooled embedding with 20 trajectory-moment features. It is subsequently used either for direct solvation free-energy prediction or to correct a descriptor-based QSAR baseline. Optional outlier-aware descriptors provide information about solvent-accessible polarity, aromatic shielding, potentially ionizable motifs, and unsupported elements. . . . 56

3.3 t-SNE projection of the 148-dimensional trajectory-derived representation. Points are colored by SMARTS-based functional-group labels for the major chemical classes. The projection shows that the trajectory-derived representation preserves chemically meaningful local structure, but the two-dimensional map is used only for visualization and not as a supervised model input or a quantitative clustering assignment. 59

3.4 Descriptor-based model performance. (A,B) R^2 and MSE values for different machine-learning models across the datasets; MSE is reported in $(\text{kcal mol}^{-1})^2$. (C-E) Predicted versus experimental ΔG_{solv} values for transfer tests in which an XGBoost model trained on MNSol is evaluated on other datasets. 62
3.5 Comparison of molecular fingerprints using XGBoost and Random Forest Regression. The upper panels show XGBoost performance and the lower panels show Random Forest performance. Bar plots report R^2 and MSE values across datasets, with MSE in $(\text{kcal mol}^{-1})^2$ and standard error bars from five-fold cross-validation. 63
3.6 Comparison between Random Forest Regression and the multilayer perceptron regressor. (A,B) Predicted versus experimental solvation free energies on CombiSolv using Morgan fingerprints. (C,D) MSE comparison across datasets and fingerprint types, reported in $(\text{kcal mol}^{-1})^2$. 64
3.7 Transferability test for CIGIN. (A–C) Predicted versus experimental aqueous solvation free energies for a model trained on MNSol and evaluated on MNSol, FreeSolv, and CombiSolv, respectively. Performance is reported using R^2 scores and MSE values, with MSE in $(\text{kcal mol}^{-1})^2$, and the decline from MNSol to the external datasets shows the effect of novel solutes under chemical-distribution shift. (D) Absolute prediction error for alkane molecules from CombiSolv as a function of carbon-chain length, showing systematically larger errors for longer alkanes; these molecules are highlighted by the dashed circle in panel C. (E,F) Cross-dataset transferability summary, where CIGIN is trained on one dataset and tested on each of the three datasets, with performance shown by MSE and R^2, respectively. (G) Predicted versus experimental ΔG_{solv} for the merged dataset under k-fold cross-validation, with points colored by functional-group classification. (H) Representative high-error molecules projected into the GCN-VAE latent space, illustrating chemically localized failure motifs that remain difficult for the graph representation. 66

4.1 Structural diversity of ice phases used in this study. Representative unit cells, oxygen-oxygen radial distribution functions, and local-neighborhood comparisons illustrate why closely related phases can be difficult to distinguish using simple geometric criteria. 72
4.2 Classical order-parameter projections for multiple ice phases. The joint $\bar{q}_3$-$\bar{q}_6$ space separates some phases but leaves substantial overlap among others. One-dimensional distributions of tetrahedrality and local entropy show a similar limitation when many phases are considered simultaneously. 74

4.3 Schematic of the *IceCoder* workflow. Oxygen-centered local environments are converted into SOAP power spectra, normalized, and passed through a fully connected VAE. The two-dimensional bottleneck coordinates, μ_1 and μ_2, provide the learned structural order-parameter space. 76
4.4 Projection of minimized ice structures into the two-dimensional VAE latent space. Most ice phases occupy distinct regions, while hydrogen-ordered counterparts appear near their corresponding disordered phases. 78
4.5 Finite-temperature ice and liquid environments in the learned latent space. (A) VAE projection of MD configurations. (B) PCA projection of the same SOAP vectors, showing stronger phase overlap. (C-E) Substructure within ice-VI, where latent-space clusters correspond to measurable differences in local geometry. 79
4.6 SVM decision regions in the $\mu_1-\mu_2$ latent space. These boundaries convert the continuous VAE projection into local phase labels for molecular simulation trajectories. 80
4.7 Hexagonal ice growth in a two-phase ice-Ih/liquid simulation. VAE projections, structure snapshots, frame-averaged latent coordinates, and comparison with CHILL+ show that *IceCoder* follows the conversion of liquid water into ice-Ih. 81
4.8 Phase evolution in a mixed ice-Ih/ice-Ic/liquid simulation. The VAE projection, structural snapshots, latent-coordinate time series, phase counts, and a single-molecule trajectory show the conversion toward cubic ice. 82

5.1 Schematic of the PathGennie sampling cycle and alanine dipeptide benchmark. (Left) The algorithm alternates between short sampler trials (blue) and conditional runner propagation (orange). (Right) Generated pathways for alanine dipeptide in the ϕ–ψ plane, illustrating how increasing the runner length τ_2 shifts trajectories from direct crossings toward minimum-free-energy routes. 88
5.2 PathGennie trajectories on the Wolfe-Quapp potential with a suboptimal progress variable. Despite guiding only along x, the algorithm discovers transitions through both the upper and lower saddle channels. Increasing the runner length τ_2 shifts the path distribution toward the lower-energy channel by allowing relaxation along the unguided y coordinate. 91
5.3 Benzene unbinding pathways from T4 lysozyme L99A. Representative trajectories from the seven identified exit route clusters are shown around the binding cavity, together with the relative population of each cluster. 92

5.4 Imatinib dissociation from Abl kinase. PathGennie identifies three main exit channels relative to the activation loop, P-loop, and αC-helix. Ligand RMSD traces (right) show the corresponding dissociation progress for representative trajectories from each channel. 93
5.5 Folding and unfolding paths for Trp-cage and Protein G. Generated trajectories show the expected change in native-contact content, and when projected onto reference free-energy surfaces constructed from long unbiased simulations, they follow the minimum-free-energy pathways. 95
5.6 Overview of the path-resolved weighted-ensemble workflow. (a) Initial reactive trajectories are generated with PathGennie. (b) Trajectories are clustered into mechanistically distinct path families using DTW. (c) For each family, a neural-network representation smooths and reparameterises the reference path. (d) The refined reference path defines PathCVs along which PathGennie is rerun until stabilisation. (e) The converged PathCV is used to discretise configuration space for pre-seeded WE simulations, yielding (f) pathway-specific kinetic observables. 96
5.7 Iterative refinement of reactive PathGennie paths. (a) Successive paths on the Müller–Brown potential approach the zero-temperature-string reference path, with source (A), target (B), and intermediate (I) annotated. (b) Discrete Fréchet distance between consecutive paths, monitoring convergence. (c) Representative view of the synthetic OAMe–G2 host–guest system; guest G2 is shown in cyan sticks. (d) PathGennie trajectories in PCA space projected onto the physical z–V_2 plane, where z is the ligand center-of-mass projection on the binding axis and V_2 is a water-coordination order parameter distinguishing wet and dry channels. (e) Iterative refinement of the wet OAMe–G2 dissociation path over seven iterations. (f) Reweighted free-energy surface from WT-MetaD using the refined wet-path PathCV, compared with the brute-force/ZTS reference pathway. 99
5.8 Rate convergence in path-informed WE versus standard source-only WE. Pre-seeding walkers along PathGennie-generated pathways accelerates the attainment of stable rate estimates for (a) alanine dipeptide and (b) chignolin. 101

5.9 Temperature-dependent pathway crossover in the three-hole potential. (a, b) Representative transition pathways at $T = 50$ K and $T = 200$ K. (c, d) Convergence of pathway-specific transition rates through the upper and lower channels at each temperature. (e) Rate estimates from WE using a suboptimal one-dimensional x coordinate, which fails to converge at low temperature. (f) Low-temperature rate convergence comparing the two-dimensional (x, y) representation with the one-dimensional coordinate. 103
5.10 Path-resolved imatinib dissociation from WT and N368S Abl kinase. Structural representations of the imatinib-bound complex for (a) WT and (b) N368S, with the mutation site in red, DFG motif in magenta, αC helix in orange, P-loop in sky blue, and activation loop in lime green. Panels (c) and (d) show the pathway-specific dissociation rates for WT and N368S, respectively, as functions of molecular time. In WT, the P-loop/hinge route carries the dominant kinetic flux; mutation redistributes this flux overwhelmingly to the αC-helix channel. 104
5.11 Path-resolved dissociation kinetics of benzamidine from trypsin (PDB 3PTB). (a) Major unbinding pathways identified from trajectory clustering shown as coloured tubes; tube radius encodes PathGennie trajectory population (not kinetic flux). The protein is shown in grey cartoon, with calcium as a green sphere. (b) Time evolution of pathway-specific dissociation rates from WE simulations. (c) Relative pathway populations in the PathGennie ensemble (colour consistent with panel (a)). Path 1 dominates the reactive flux by more than an order of magnitude over Path 2, and Path 3 is negligible. 106

6.1 Schematic workflow of TRAILS-MD. A YAML configuration file defines the molecular system, MD engine, sampling space, spawning rule, and checkpointing options. The core driver prepares the requested engine and launches an ensemble of short parallel walker trajectories. Generated frames are converted into fixed collective variables or molecular features; in adaptive mode, PCA, TICA, TVAE, or Deep-TICA models are trained or updated at scheduled intervals, and historical features are reprojected when the latent representation changes. Spawning is performed on the accumulated projected history, after which the selected frames are mapped back to full-system coordinates to initialize the next walker ensemble. At each iteration, the framework records trajectory history, parent–child lineage, checkpoint state, occupancy diagnostics, and run logs, enabling restartable and fully traceable adaptive sampling campaigns. 115

6.2 Adaptive sampling of alanine dipeptide in the Ramachandran plane. Representative structures corresponding to the (a) C_7^{eq}, (b) α_R, (c) α_L, and (d) C_7^{ax} conformations. (e) Cumulative conformational distribution sampled by TRAILS-MD projected onto the (ϕ, ψ) plane; the color scale reports the negative logarithm of the estimated probability density, and white crosses mark the four reference conformations. (f) Cumulative grid coverage as a function of adaptive iteration for density-based and Voronoi-based spawning, both evaluated on the same 36×36 ϕ/ψ grid. 125

6.3 Adaptive sampling of CLN025 in physical and learned coordinate spaces. (a) Configurations generated using fixed physical CVs (R_g and RMSD), with spawning performed in the same plane. (b, c) Physical R_g–RMSD projections of configurations generated while spawning in learned two-dimensional TVAE and TICA spaces, respectively. All three panels use the same physical axes to enable direct ensemble comparison. (d) Representative conformations A–E selected from the explored ensemble, spanning compact hairpin-like, partially structured, helical, and extended states. (e) Sampled density in the final TVAE latent space, colored by the relative free-energy-like landscape ($-RT \ln p(\mathbf{x})$) estimated by reweighting with a maximum-likelihood Markov state model rather than directly from the adaptive walker counts. 127

6.4 Endpoint and torsional analysis of AIB9. (a) Representative left- and right-handed helical conformations of AIB9. (b) Locations of the endpoint conformations in the ϕ_5–ψ_5 torsional projection; the background density-derived landscape is shown as the negative logarithm of the estimated probability density. (c, d) TRAILS-MD sampling in the ϕ_5–ψ_5 and ϕ_8–ψ_8 torsional planes from simulations performed in the four-dimensional torsional space, illustrating endpoint coverage without demonstrating a continuous transition. (e, f) A continuous L-to-R transition trajectory reconstructed by tracing the ancestry of walkers generated in the VAE+LDA projection space, shown in the (ϕ_5, ψ_5) torsional and R_g–RMSD planes (RMSD computed with respect to the right-handed reference R state). . . 129

6.5 Adaptive sampling of two fast-folding proteins. (a) Trp-cage (2JOF) sampled in the RMSD–fraction-of-native-contacts plane, showing exploration from the unfolded ensemble toward the folded basin. (b) WW domain sampled using a three-dimensional TICA space learned from unfolding trajectories and visualized in the physical R_g–fraction-of-native-contacts plane. Representative cartoons are shown within each panel, and relative density is colored by $-RT \ln \hat{p}(\mathbf{x})$. 131

6.6 Exploratory ligand-unbinding applications of TRAILS-MD. (a) Bound structure of the OAMe-G2 host–guest complex. (b) Fraction of the configured two-dimensional sampling space explored as a function of adaptive iteration for OAMe-G2. (c) Distribution of sampled configurations in the projected host–guest unbinding space, showing the spread of explored positions and emergence of multiple escape directions. (d) Bound N-HSP90α–ligand complex (PDB 6EI5). (e) Superposition of B5Q center-of-mass positions sampled during the TRAILS-MD campaign, providing a qualitative map of possible ligand-exit tunnels around the binding site. The green bounding box denotes the center-of-mass region used to define the TRAILS-MD sampling space. 133
6.7 MSM construction from TRAILS-MD-seeded trajectories. (a) CLN025 conformational space explored by TRAILS-MD, projected in the RMSD–FNC plane, with selected spawn points marked. (b) MSM-reweighted sampled density in the same physical projection, constructed from production trajectories initialized from the selected spawn structures. (c) Implied timescales as a function of MSM lag time, used to assess Markovian behavior and guide lag-time selection. (d) Two-state coarse-grained MSM with estimated transition probabilities between the folded and unfolded states (illustrated by arrows). 134
6.8 Runtime decomposition and multi-GPU scaling of TRAILS-MD for the WW-domain benchmark. (a) Average wall time per adaptive iteration using one to four GPUs, decomposed into MD execution time and non-MD analysis overhead. Numerical annotations above the bars report the overhead fraction of total iteration time in percent. (b) Observed speedup of the MD execution component relative to the one-GPU baseline. The dashed line indicates ideal linear scaling, and values next to markers report corresponding parallel efficiency. Benchmark: 16 parallel walkers of 100 ps each per iteration, GROMACS 2022.4, NVIDIA GeForce RTX 2080 Ti. 135

List of Tables

3.1 Progressive gains and remaining transfer limitations across the four representation families. Each representation resolves part of the preceding limitation, but no representation eliminates all forms of chemical-distribution shift. MAE values are reported in kcal mol^{-1}, whereas MSE values are reported in $(kcal\,mol^{-1})^2$. 61

3.2 Transfer performance of the controlled trajectory-informed models. The CombiSolv test set contains 254 held-out molecules after leakage-aware filtering; MNSol is interpreted as a 65-molecule external diagnostic stress test. MAE values are reported in kcal mol^{-1} with nonparametric bootstrap 95% confidence intervals estimated from the molecule-level prediction errors. 67

4.1 VAE architecture and training hyperparameters for *IceCoder*. 76

4.2 Thermodynamic conditions and simulation cell dimensions used to generate homogeneous ice-phase trajectories. 77

5.1 Representative PathGennie parameters and computational costs. Reactive length denotes the cumulative time along accepted productive segments; total length includes unproductive trials. Wall times correspond to single-trajectory costs on one NVIDIA RTX 2080 Ti GPU. 91

5.2 Path-specific imatinib dissociation rates from WT and N368S Abl kinase estimated with the WE-PathCV workflow, compared with infrequent metadynamics **[Shekhar2022]** and experiment **[Lyczek2021]**. WE-PathCV uncertainties are Bayesian-bootstrap 95% confidence intervals over three independent replicates in the converged analysis window. Aggregate WE simulation cost per path: $\approx 5.3\ \mu s$. 105

5.3 Path-specific benzamidine–trypsin unbinding rates from WE-PathCV simulations, evaluated over the converged 14–15 ns molecular-time window. Bayesian-bootstrap 95% CI from three independent WE replicates. Aggregate WE simulation cost per path: $\approx 2.0\ \mu s$. 107

5.4 Comparison of k_{off} estimates and aggregate simulation costs for benzamidine–trypsin dissociation from various computational approaches. 108

1

Introduction

Molecular science asks how atomic-scale interactions become observable behavior in materials, biological assemblies, and chemical processes. In aqueous solvation, many transient solvent contacts combine to produce a measurable solvation free energy. In a supercooled liquid, a small region of local order may either dissolve within a few picoseconds or continue to grow into a crystalline nucleus. Ligand binding and protein folding pose an additional difficulty: the relevant motions are rare events in which thermal fluctuations must locate a viable route through a high-dimensional landscape before a macroscopic transition is observed.

Although these problems arise in different areas of molecular science and span widely separated spatial and temporal scales, they repeatedly lead to the same computational progression. One begins with microscopic coordinates, constructs a representation that can be simulated or learned from, and then seeks a reduced description in terms of molecular properties, metastable states, phases, order parameters, or collective variables (CVs). This reduced description can subsequently be used to guide sampling towards regions that conventional molecular dynamics (MD) would reach only after prohibitively long waiting times. In this thesis, that progression can be summarized as a continuous chain from *molecular coordinates to representations, predictive or generative models, physically meaningful states or collective variables, enhanced sampling, and ultimately molecular discovery*. This sequence provides the conceptual backbone of the dissertation.

Three bottlenecks recur along that path, with system-specific manifestations:

1. **The representation bottleneck.** A molecular system must be encoded in a form that retains the relevant physics while remaining computationally tractable. Each choice, from raw Cartesian coordinates to internal coordinates, descriptors, fingerprints, molecular graphs, or machine-learned latent variables, preserves different information and introduces distinct biases.

2. **The identification bottleneck.** High-dimensional simulation data must be converted into physically meaningful states, order parameters, phases, or CVs. This task becomes difficult when metastable states overlap in simple projections, when several distinct mechanisms share the same endpoint, or when the relevant slow coordinate is unknown before sampling.

3. **The sampling bottleneck.** Transitions of interest, including ligand unbinding, protein folding, phase transitions, and activated conformational changes, often occur on microsecond-to-second timescales or longer, whereas atomistic simulation requires femtosecond integration steps. Standard MD therefore spends most of its computational effort within metastable basins and only rarely samples the transitions that determine thermodynamics and kinetics.

This thesis treats prediction, identification, and sampling as connected aspects of a single representation problem. Chapter 3 asks whether molecular representations transfer when predicting aqueous solvation free energy, or whether their apparent accuracy is restricted to the data distribution on which they were trained. The chapter follows a progression from descriptors and fingerprints to graph models and trajectory-derived embeddings, while testing whether conformational information and explicit applicability-domain descriptors improve prediction under chemical shift. Chapter 4 narrows the focus from whole molecules to local water environments, using Smooth Overlap of Atomic Positions (SOAP) descriptors and a variational autoencoder (VAE) to distinguish ice phases in simulation data for which classical order parameters are limited. The last two chapters address rare events directly: Chapter 5 combines pathway generation, path clustering, path collective variables (PathCVs), and Weighted Ensemble (WE) kinetics, and Chapter 6 develops *TRAILS-MD*, an adaptive sampling framework in which fixed or learned latent spaces guide the launch points of subsequent trajectory rounds.

1.1 Molecular Modeling Across Scales

Molecular modeling supplies the microscopic account behind macroscopic observables. Classical thermodynamics describes equilibrium systems through state variables such as temperature, pressure, entropy, and free energy, but it does not specify how these quantities arise from atomic interactions. Molecular simulation makes that connection explicit by constructing models of atomic interactions and using them to generate ensembles of configurations from which observables can be estimated.

The examples that motivate this thesis do not live at one level of description. Solvation free energy depends on molecular chemistry and solvent response. Ice nucleation depends on local structural order and phase identity. Ligand unbinding, conformational change, and folding depend on rare motion through complex free-energy landscapes. For these systems, a force field and an integrator are only the starting point. The harder choices are what structure to represent, which states or coordinates are worth tracking, and how to reach configurations that ordinary trajectories may almost never visit.

1.2 Statistical Mechanics and Molecular Dynamics

Statistical mechanics formalizes the connection between microscopic configurations and macroscopic observables by relating thermodynamic quantities to averages over phase space.

For a system of N particles with coordinates $\mathbf{r}$, momenta $\mathbf{p}$, and Hamiltonian $H(\mathbf{r}, \mathbf{p})$, the canonical partition function is

$$Q = c \int e^{-\beta H(\mathbf{r},\mathbf{p})} d\mathbf{r}\, d\mathbf{p}, \tag{1.1}$$

where $\beta = 1/k_B T$, k_B is the Boltzmann constant, and $c = 1/(N! h^{3N})$ accounts for particle indistinguishability and Planck's constant. These relations follow standard statistical-mechanical treatments of the canonical ensemble [**Frenkel2001**]. The Helmholtz free energy follows directly:

$$A = -k_B T \ln Q. \tag{1.2}$$

In principle, this expression determines the equilibrium thermodynamics of the chosen molecular model in the canonical ensemble. In practice, the integral extends over an enormous phase space and cannot be evaluated analytically for realistic molecular systems.

Molecular dynamics replaces direct integration over phase space with time averaging along simulated trajectories. Under the ergodic hypothesis, the ensemble average of an observable is recovered in the infinite-time limit:

$$\langle O \rangle = \lim_{t_{\text{sim}} \to \infty} \frac{1}{t_{\text{sim}}} \int_0^{t_{\text{sim}}} O(\mathbf{r}(t), \mathbf{p}(t))\, dt. \tag{1.3}$$

Equation 1.3 explains why MD is useful and why it fails in finite time. A trajectory can estimate an observable only to the extent that it visits the basins that contribute to that observable. If it remains trapped, the average is a property of the sampled subset rather than of the full ensemble.

The next question is what molecular model generates those trajectories, and what representation of the resulting configurations is retained for comparison, learning, or sampling.

1.3 Molecular Models and the Resolution–Efficiency Tradeoff

The accuracy and scope of a molecular simulation are determined by the physical model used to represent interactions. At one end of the hierarchy are electronic-structure

methods, which compute energies from the electronic Schrödinger equation. Density functional theory (DFT) and post-Hartree–Fock methods can capture bond breaking, charge transfer, electronic polarization, and chemical reactivity [**Hohenberg1964**, **Kohn1965**, **Raghavachari1989**]. Their cost, typically scaling from roughly cubic to much higher polynomial cost depending on the electronic-structure method, restricts routine simulations to small systems and short timescales. *Ab initio* MD can propagate trajectories on DFT-level surfaces, but practical applications are commonly limited to hundreds of atoms and picosecond-scale trajectories.

Classical force fields take the complementary approach. They replace explicit electronic structure with analytical functions of nuclear coordinates:

$$V(\mathbf{r}) = \sum_{\text{bonds}} \frac{1}{2} k_b (r - r_0)^2 + \sum_{\text{angles}} \frac{1}{2} k_\theta (\theta - \theta_0)^2 + V_{\text{dihedral}} + V_{\text{vdW}} + V_{\text{elec}}. \tag{1.4}$$

This approximation enables simulations of solvated proteins, materials, and chemical assemblies over nanosecond to microsecond timescales. The cost is a restricted functional form: fixed partial charges, harmonic bonded terms, approximate dispersion and electrostatics, and limited treatment of chemical reactivity. Force-field accuracy is therefore tied to both the training data used for parameterization and the representation chosen for the molecular system.

Coarse-grained and continuum models move further along the accuracy–efficiency spectrum by grouping atoms into interaction sites or replacing particles with fields. These models extend accessible length and time scales, but discard microscopic detail. The practical question is not whether one model is universally superior, but which level of representation is sufficient for the scientific question being asked.

Figure 1.1 provides the resolution map used throughout the thesis. Later chapters repeatedly move along the same axis: keep enough molecular detail to preserve the relevant physics, while compressing the system enough to predict, classify, or sample it.

The history of MD can be read as a steady expansion of what could be represented and for how long. Alder and Wainwright showed with hard-sphere dynamics that even simple repulsive particles could exhibit a liquid–solid phase transition [**Alder1957**]. Rahman's liquid-argon simulation then replaced that discontinuous picture with a continuous Lennard-Jones potential and connected classical trajectories to experimental structure and transport [**Rahman1964**]. With stable integrators such as Verlet's algorithm [**Verlet1967**], simulations could move toward more realistic molecular liquids, including water [**Stillinger1971**]. Biomolecular force fields [**Lifson1968**, **Levitt1975**] extended that logic to proteins, culminating early on in the bovine pancreatic trypsin inhibitor simulation of McCammon, Gelin, and Karplus [**McCammon1977**].

By the 1990s, packages such as AMBER, CHARMM, and GROMACS made biomolec-

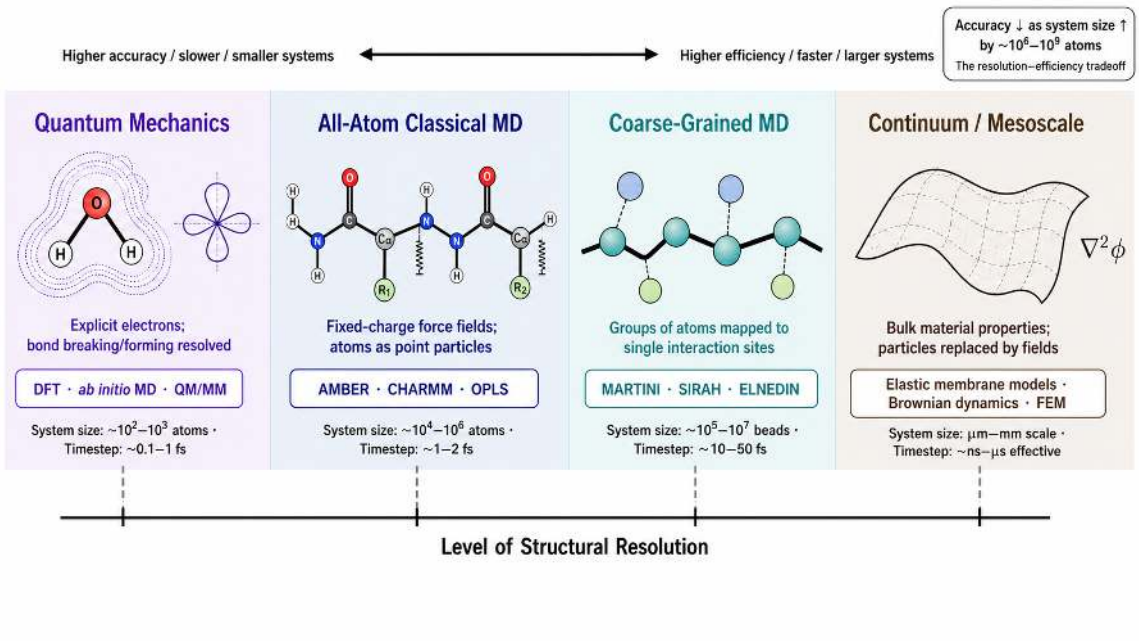

Figure 1.1: Resolution hierarchy of molecular modeling approaches. Quantum mechanical descriptions treat electrons explicitly and can resolve chemical rearrangement; all-atom classical MD replaces electrons with empirical interaction functions; coarse-grained models map groups of atoms to effective particles; and continuum descriptions replace molecular detail with field-level variables. The same ordering underlies the later choice of compact descriptors, local structural representations, and path-based sampling coordinates.

ular simulation broadly accessible [**Pearlman1995**, **Brooks1983**, **Berendsen1995**]. Particle Mesh Ewald methods improved long-range electrostatics in periodic systems [**Darden1993**]. GPU acceleration and specialized hardware such as Anton extended the reachable timescale by orders of magnitude [**Anderson2008**, **Shaw2008**]. Millisecond-scale simulations demonstrated that brute-force MD can capture slow conformational transitions in selected systems [**Shaw2010**]. Even with these gains, brute-force MD does not remove the barrier problem: transition times still grow exponentially with free-energy barriers. The sampling chapters begin from that constraint.

1.4 The Timescale and Rare-Event Problem

Atomistic MD is governed by two very different clocks. The fast one is set by bonded vibrations: hydrogen-containing bonds usually require timesteps of approximately 1–2 fs for stable integration. The slow one is set by the events of interest. Ligand dissociation, protein folding, conformational switching, crystal nucleation, and phase transformation may occur on microsecond, millisecond, or longer timescales. The simulation must resolve the first clock even when the scientific question lives on the second.

As an illustrative estimate, a GPU-accelerated simulation producing 10^6 timesteps per second with a 2 fs timestep would generate roughly 2 ns of simulated time per second of wall-clock time. A millisecond trajectory would require approximately 5×10^5 seconds of continuous computation for a single trajectory, before accounting for replicas, equilibration, analysis, or force-field sensitivity tests. For second-scale events, direct simulation is generally infeasible.

Figure 1.2 puts these two clocks on the same axis, from femtosecond integration steps to transitions that may require microseconds or longer to observe.

Transition State Theory (TST) expresses the same problem in energetic terms [**Eyring1935**]. For a free-energy barrier $\Delta G^{\ddagger}$, the rate constant is

$$k = \kappa \frac{k_B T}{h} \exp\left(-\frac{\Delta G^{\ddagger}}{k_B T}\right), \tag{1.5}$$

where κ is the transmission coefficient and h is Planck's constant. Because the rate depends exponentially on the barrier height, modest increases in $\Delta G^{\ddagger}$ can shift a process from nanoseconds to milliseconds or seconds. Standard MD trajectories therefore spend most of their computational budget fluctuating inside metastable basins rather than crossing the barriers that determine kinetics.

The finite-time consequence is not a failure of the formal ergodic hypothesis, but a

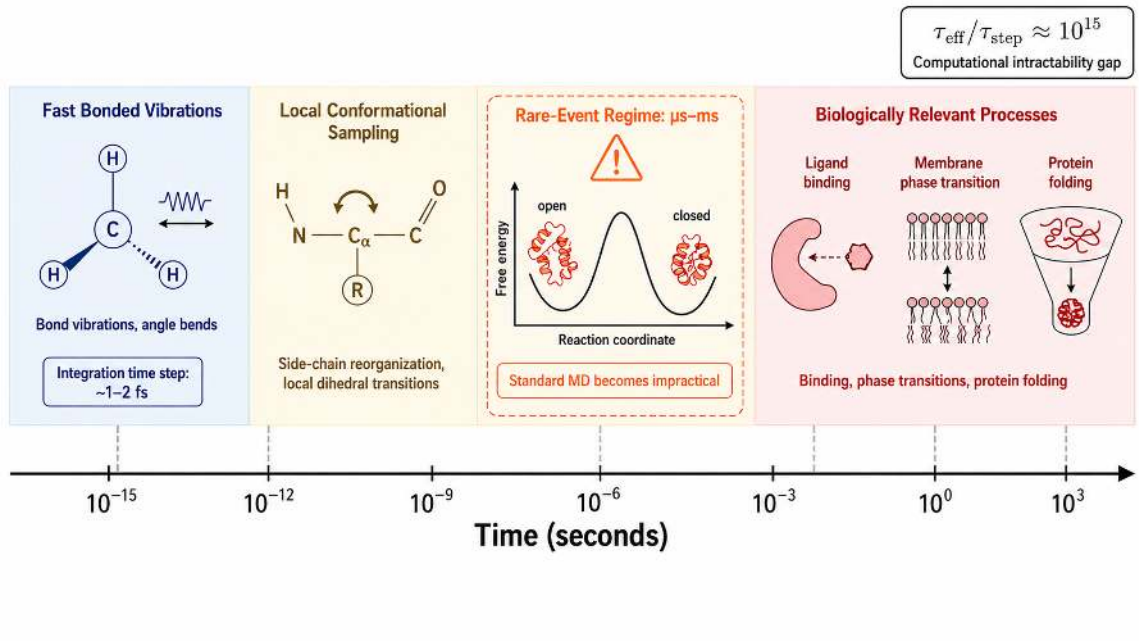

Figure 1.2: The timescale gap in molecular simulation. Femtosecond timesteps are required to integrate fast bonded vibrations, while relevant transitions such as ligand binding, conformational switching, nucleation, and folding may occur on microsecond-to-second scales. Enhanced sampling and adaptive simulation strategies are responses to this separation.

practical failure of convergence on accessible simulation timescales:

$$O_{\text{sim}} = \frac{1}{t_{\text{sim}}} \int_0^{t_{\text{sim}}} O(t)\, dt \neq \langle O \rangle. \tag{1.6}$$

If t_{sim} is shorter than the characteristic transition time between basins, the average is biased toward the initial state. The problem is not in the analysis step: thermodynamic or kinetic information cannot be recovered from regions of configuration space that were never visited.

Chapters 5 and 6 address this limitation from complementary directions. Chapter 5 treats the transition pathway itself as the object to organize, using PathGennie and path-resolved WE to focus effort on route-specific kinetics. Chapter 6 relaxes the assumption that the relevant coordinates or endpoints are already known, and uses TRAILS-MD to broaden exploration through adaptive spawning.

1.5 The Dimensionality and Representation Problem

The sampling bottleneck is inseparable from the dimensionality of molecular configuration space. A system of N atoms has $3N$ Cartesian coordinates. A solvated biomolecule with 50,000 atoms therefore occupies a 150,000-dimensional coordinate space before momenta are considered. Exhaustive exploration of such a space is impossible, and the

number of samples required to cover it at fixed resolution grows exponentially with dimensionality.

The difficulty is not only the number of coordinates. Most molecular coordinates do not contribute equally to the slow process being studied. Fast local vibrations may equilibrate quickly, while the motion that controls thermodynamics or kinetics lies on a much lower-dimensional structure embedded in the full coordinate space. A useful representation must preserve that slow structure without carrying every irrelevant fluctuation along with it.

The transfer-operator formalism gives this statement a dynamical form. A propagator $\mathcal{T}(\tau)$ evolves the probability density $\rho(\mathbf{r})$ over lag time τ and has eigenfunctions associated with the slow relaxation modes:

$$\mathcal{T}(\tau)\psi_i = \lambda_i(\tau)\psi_i, \tag{1.7}$$

where $\lambda_i(\tau) = e^{-\tau/t_i}$ and t_i are implied timescales. The leading eigenfunctions provide optimal coordinates for slow relaxation in the variational sense, although they are generally unknown for realistic molecular systems and must be estimated from data [**Noe2017**]. This transfer-operator view has also been operationalized as a learning objective: VAMPnets train neural networks to approximate dominant dynamical modes from trajectory data [**Mardt2018**], while graph-based extensions apply related variational principles to learn slow collective variables for self-assembly and other many-body processes without hand-crafted fixed-order input features [**Liu2023GraphVAMPnets**]. The practical problem is circular. The slow coordinates are easiest to learn from trajectories that have already crossed barriers, but those crossings are exactly what good slow coordinates are meant to help obtain.

The representation problem takes a different form in each application. In solvation, descriptors, fingerprints, molecular graphs, and trajectory-derived embeddings must retain the physical factors governing solvation free energy, including molecular size, solvent-accessible polarity, conformational exposure, and applicability-domain limits. In ice-phase identification, SOAP descriptors and a VAE latent space must preserve local structural distinctions among water environments. In pathway generation and adaptive sampling, the harder requirement is to capture progress through rare transitions without erasing route identity, whether the coordinate is a CV, PathCV, TICA coordinate, TVAE latent variable, or Voronoi-spawning space.

Figure 1.3 shows this spectrum of molecular representations. It also marks the link between static prediction, structural identification, and sampling.

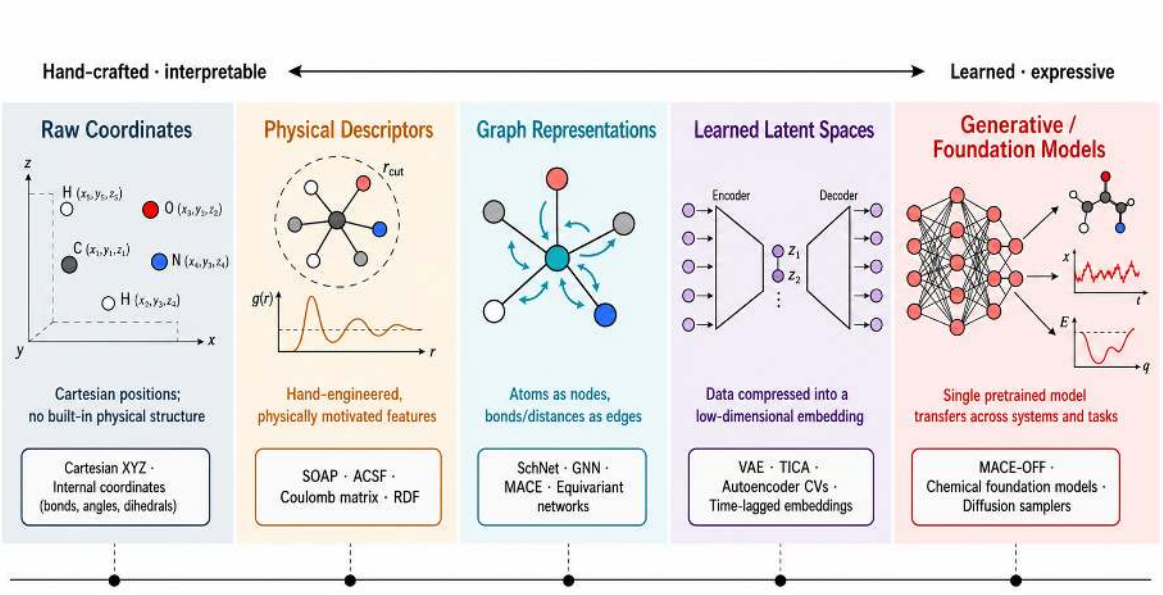

Figure 1.3: Spectrum of molecular representations used in molecular modeling and machine learning. Raw coordinates preserve direct geometry but are high-dimensional; descriptors and fingerprints provide compact hand-crafted encodings; molecular graphs preserve chemical topology; trajectory-derived embeddings summarize conformational and physical information from molecular simulations; local descriptors such as SOAP encode symmetry-aware environments; and learned latent spaces compress molecular data into variables useful for classification, visualization, and sampling.

1.6 Collective Variables and the Identification Bottleneck

A collective variable (CV) is a low-dimensional function

$$\boldsymbol{\xi}(\mathbf{r}) : \mathbb{R}^{3N} \rightarrow \mathbb{R}^{d}$$

that maps atomic coordinates onto a reduced space intended to describe a molecular process. CVs are used to visualize free-energy landscapes, define metastable states, construct order parameters, bias simulations, bin trajectories, and monitor transition progress. A useful CV should distinguish relevant metastable states, capture slow dynamics, be sufficiently continuous for analysis or biasing, and retain enough physical interpretability to support mechanistic conclusions.

Traditional CVs are hand-crafted from physical intuition. Common examples include interatomic distances, ligand–protein separations, backbone dihedral angles, root-mean-square deviation (RMSD), radius of gyration, native-contact fractions, Steinhardt bond-order parameters, tetrahedrality, and local entropy. These variables remain valuable because they are interpretable and inexpensive. Chapter 4, for example, compares learned ice-phase representations against classical order parameters such as tetrahedrality and Steinhardt descriptors.

The limitation is that hand-crafted CVs can be incomplete or degenerate. Distinct physical states may collapse onto the same CV value, producing hidden barriers in projected free-energy surfaces. Multi-pathway processes are especially difficult: a ligand may unbind through several protein tunnels with similar ligand–protein distances, or a phase transition may proceed through local structures that overlap in simple order-parameter space. In such cases, the CV may describe endpoint progress but fail to preserve pathway identity.

The operational difficulty is circular. Efficient rare-event sampling usually needs a low-dimensional coordinate that captures the transition, but learning that coordinate from data usually requires trajectories that already contain the transition. This circular dependence is a central difficulty in enhanced sampling **[Valsson2014VES]**. Chapter 6 confronts it directly, because adaptive sampling must update its representation while using that same representation to decide where new simulations should start. Chapter 5 faces the same issue from the path side, where PathGennie and PathCV refinement use imperfect initial progress coordinates to discover and stabilize physically meaningful pathways.

The same issue reappears in state identification. A classifier or latent space can only be trusted if it separates physically meaningful states rather than artifacts of the descriptor or training data. For this reason, Chapter 4 validates the learned SOAP–VAE coordinates by cluster separation and by their ability to track phase evolution in

two-phase and three-phase simulations.

Recent software efforts show that learned CVs are becoming easier to use in practice: unified libraries now allow architectures based on dimensionality reduction, metastable-state classification, and slow-mode objectives to be trained and deployed through common interfaces coupled to enhanced-sampling engines [**Bonati2023Mlcolvar**]. Recent perspectives have also reframed CV selection as a spectrum from human-intuited coordinates to fully learned ones, rather than a binary choice between the two [**Fu2024CVEnhancedSampling**].

1.7 Enhanced Sampling Methods

Enhanced sampling methods address rare events by changing where simulation effort is spent. Some methods raise the probability of barrier crossing, some broaden coverage of configuration space, and others concentrate computation on reactive trajectories. They do not remove the need for physical validation; rather, they shift the problem toward choosing appropriate representations, coordinates, interfaces, bins, or adaptive rules. The integration of ML into enhanced sampling has become substantial enough to warrant dedicated review, with recent surveys cataloguing dimensionality reduction, reinforcement learning, and flow-based approaches as major emerging strategies [**Mehdi2024EnhancedSamplingML**].

Figure 1.4 organizes the main method families by the part of the simulation problem they act on. The categories are not strict boundaries, but they clarify how the methods developed in this thesis relate to established approaches.

1.7.1 Collective-Variable Biasing

CV-biasing methods modify the effective potential along selected coordinates. Umbrella sampling applies harmonic restraints in overlapping windows and reconstructs unbiased free-energy profiles using WHAM [**Kumar1992**] or MBAR [**Shirts2008MBAR**]; the original formulation traces to Torrie and Valleau [**Torrie1977**]. Metadynamics deposits history-dependent bias potentials along CVs to discourage revisiting already sampled regions [**Laio2002**]; well-tempered metadynamics improves convergence by tempering the deposited bias [**Bussi2006**]. Variationally enhanced sampling and OPES recast bias construction in probabilistic or variational terms [**Valsson2014VES**, **Invernizzi2020OPES**]. Adaptive Biasing Force (ABF) methods estimate and compensate the mean force along a coordinate [**Darve2001**].

These methods can produce accurate free energies when the CVs are appropriate. Their main vulnerability is omitted slow motion. If an important coordinate is orthogonal to the biased variables, the simulation may exhibit hysteresis, slow convergence, or misleading projected free energies. The learned and path-based representations used

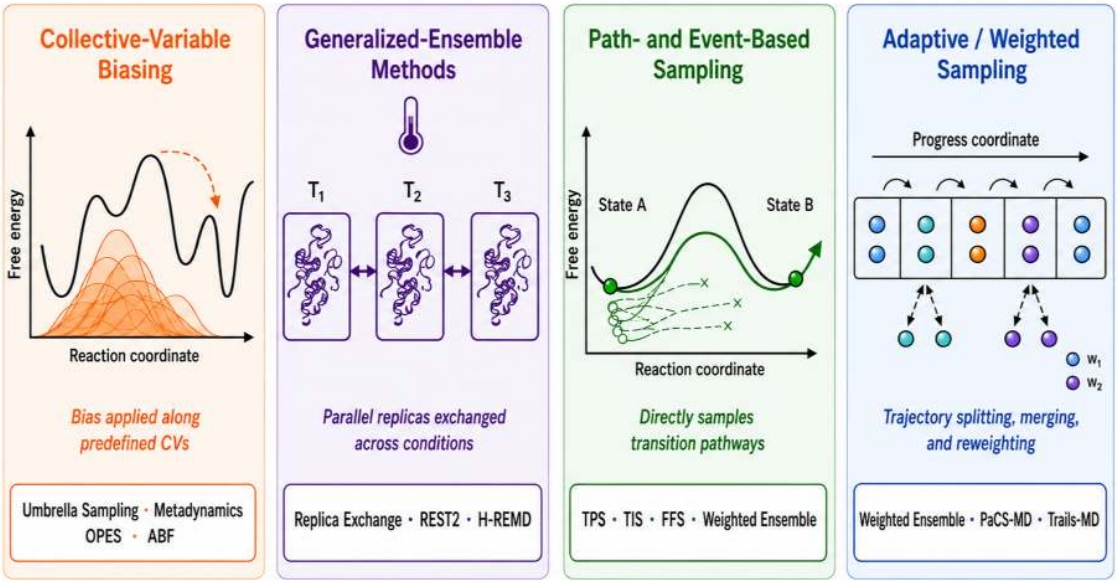

Figure 1.4: Classification of enhanced sampling methods. CV-biasing methods alter the effective potential along selected coordinates; generalized-ensemble methods improve exploration through thermodynamic or Hamiltonian exchange; path- and event-based methods sample transition trajectories or interfaces; Weighted Ensemble uses statistically weighted splitting and merging of trajectory ensembles; and adaptive or ML-guided methods update sampling decisions using learned or evolving representations.

later in the thesis are aimed at this failure mode.

1.7.2 Generalized-Ensemble Methods

Generalized-ensemble approaches accelerate barrier crossing by running simulations across multiple thermodynamic states or Hamiltonians. Temperature replica exchange MD (T-REMD) runs replicas at different temperatures and exchanges configurations with an acceptance probability satisfying detailed balance [**Sugita1999**]:

$$P_{\text{acc}} = \min\left(1, \exp\left[(\beta_i - \beta_j)(U_i - U_j)\right]\right), \tag{1.8}$$

where β_i and β_j are inverse temperatures and U_i, U_j are potential energies. Hamiltonian replica exchange varies interaction parameters rather than temperature [**Fukunishi2002**], while REST/REST2 temper selected solute interactions to reduce the replica count for solvated biomolecules [**Liu2005REST**].

Replica methods reduce reliance on specific CVs, but their cost increases with system size because sufficient exchange rates require overlapping energy distributions [**Sindhikara2008**]. They are therefore powerful for many equilibrium questions but may be expensive for large solvated systems and do not automatically provide pathway-resolved kinetic decomposition.

1.7.3 Path-Based, Event-Based, and Weighted-Ensemble Methods

Path-based methods focus on the reactive trajectory ensemble. Transition Path Sampling (TPS) performs Monte Carlo moves in trajectory space to sample paths connecting reactant and product basins [**Dellago1998**, **Bolhuis2002**]. Transition Interface Sampling (TIS) and Forward Flux Sampling (FFS) decompose transitions into crossings of interfaces along a progress coordinate [**vanErp2003**, **Allen2005**, **Allen2006**]. Milestoning partitions a reaction into transitions among milestones and reconstructs long-time kinetics from local statistics [**Faradjian2004**].

Weighted Ensemble (WE) sampling occupies an important position in this family. WE maintains a collection of trajectories with statistical weights and periodically splits or merges walkers within bins along a progress coordinate [**Huber1996**]. Because the underlying dynamics are not biased, WE can estimate rates from probability fluxes while preserving rigorous statistical weights. Its efficiency, however, depends strongly on the binning strategy and progress coordinate. Poor bins can waste walkers in unproductive regions or merge trajectories that belong to different mechanisms.

This framework sets up Chapter 5, where PathGennie-generated pathways are converted into PathCVs and used to initialize path-resolved WE simulations.

1.7.4 Adaptive and ML-Guided Sampling

Adaptive sampling changes the allocation of simulation effort based on accumulated data. Instead of running one long trajectory, many short trajectories are launched, analyzed, and restarted from selected configurations. The selection rule may prioritize low-density regions, high uncertainty, target progress, structural diversity, or slow-coordinate coverage.

ML-guided adaptive sampling extends this idea by updating the representation itself. PCA, TICA, VAEs, time-lagged autoencoders, VAMPnets, Deep-TICA, and related models can project high-dimensional features into lower-dimensional latent spaces [**PerezHernandez2013tICA**, **Wehmeyer2018**, **Hernndez2018**, **Mardt2018**, **Bonati2021**]. New simulations can then be spawned from boundaries, sparse regions, or outliers in these spaces. Chapter 6 develops this principle in *TRAILS-MD*, where fixed CVs, TICA, TVAE, and other learned projections can be coupled to density or Voronoi-based spawning while preserving trajectory lineage.

Enhanced-sampling methods differ mainly in what they modify: the potential, the thermodynamic state, the trajectory ensemble, or the allocation of new simulations. CV-biasing and generalized-ensemble methods alter the sampling distribution directly; path-based and WE methods organize transition trajectories; adaptive methods change where future simulation effort is spent. Recent ML work enters at each of these points, most often by learning coordinates, latent spaces, or selection rules that make the sampling decision less hand-tuned.

Within CV biasing, classical approaches such as umbrella sampling, WHAM, MBAR, and the adaptive biasing force estimate free-energy profiles along pre-defined coordinates [**Torrie1977**, **Kumar1992**, **Shirts2008MBAR**, **Darve2001**]. Metadynamics and its well-tempered, infrequent, bias-exchange, and parallel-tempering variants construct adaptive history-dependent biases to discourage revisiting sampled regions [**Laio2002**, **Barducci2008**, **Tiwary2013InMetaD**, **Piana2007BiasExchange**, **Bussi2006**]. Variational and data-driven reformulations including VES, OPES, SGOOP, and RAVE connect enhanced sampling to learned collective variables [**Valsson2014VES**, **Invernizzi2020OPES**, **Tiwary2016SGOOP**, **Ribeiro2018**]. Generalized-ensemble methods including replica exchange [**Sugita1999**, **Fukunishi2002**], multicanonical sampling [**Berg1991Multicanonical**], simulated tempering [**Marinari1992SimulatedTempering**], Wang–Landau sampling [**Wang2001WangLandau**], REST/REST2 [**Liu2005REST**, **Wang2011**], replica-exchange umbrella sampling [**Sugita2000REUS**], and integrated tempering sampling [**Gao2008ITS**] accelerate barrier crossing by simulating multiple thermodynamic or Hamiltonian states. Boost-potential methods including hyperdynamics, accelerated MD (aMD), and Gaussian accelerated MD (GaMD) and its ligand, peptide, and

second-generation variants (LiGaMD, Pep-GaMD, LiGaMD2) smooth or boost the potential-energy surface to reduce barriers without requiring predefined collective variables [**Voter1997Hyperdynamics**, **Hamelberg2004AMD**, **Miao2015GaMD**, **Miao2020**, **Wang2020PepGaMD**, **Wang2023LiGaMD2**].

Path-based methods focus on the reactive trajectory ensemble. Transition Path Sampling (TPS) performs Monte Carlo moves in trajectory space [**Dellago1998**, **Bolhuis2002**], while Transition Interface Sampling (TIS) and Forward Flux Sampling (FFS) decompose transitions into interface crossings [**vanErp2003**, **Allen2005**, **Allen2006**]. Milestoning partitions the reaction coordinate into milestones [**Faradjian2004**]. Minimum-free-energy-path methods such as the nudged elastic band (NEB) and the string and finite-temperature string methods identify low-energy transition routes [**Henkelman2000**, **E2002**, **Maragliano2006FTS**]. Committor optimization and transition-path theory characterize reaction mechanisms, and path collective variables (PathCVs) project trajectory data onto progress and distance coordinates [**Peters2006Committor**, **E2006TPT**, **Branduardi2007**]. Aimless shooting and ML-guided path sampling (AIMMD) automate transition-path discovery [**Peters2006AimlessShooting**, **Jung2023AIMMD**]. The Weighted Ensemble (WE) method and its steady-state, NEUS, WESTPA, WExplore, REVO, weighted-ensemble milestoning, and haMSM extensions maintain statistically weighted trajectory copies for quantitative rate estimation [**Huber1996**, **Bhatt2010**, **Warmflash2007NEUS**, **Zwier2015**, **Russo2022WESTPA2**, **Dickson2014**, **Donyapour2019**, **Ray2018WEMilestoning**, **Copperman2020**].

Adaptive sampling generalizes iterative trajectory exploration through MSM-guided seeding [**Swope2004MSM**, **Noe2009MSM**, **Prinz2011**], FAST [**Zimmerman2015**], ExTASY [**Balasubramanian2016ExTASY**], DeepDriveMD [**Lee2019**], REAP reinforcement learning [**Shamsi2018**], and uncertainty-aware selection [**Hruska2018Adaptive**]. ML-learned latent spaces including TICA [**PerezHernandez2013tICA**, **Schwantes2013TICA**], diffusion maps [**Coifman2006DiffusionMaps**, **Nadler2006DiffusionMaps**], autoencoder-based CVs [**Wehmeyer2018**, **Hernndez2018**], VAMPnets [**Mardt2018**], GraphVAMPnets [**Liu2023GraphVAMPnets**], Deep-TICA [**Bonati2021**], Deep-LDA [**Bonati2020**], SPIB [**Wang2019SPIB**], mlcolvar [**Bonati2023Mlcolvar**], and Boltzmann generators [**Noe2019BoltzmannGenerators**] provide low-dimensional representations that capture slow dynamics and can be coupled to adaptive or biased sampling. Chapter 6 develops this principle in TRAILS-MD, where fixed or learned representations guide lineage-aware adaptive spawning.

1.8 Machine Learning in Molecular Modeling and Simulation

Machine learning is useful here only when it is tied to a physical task. In this thesis, ML is used to test molecular representations for property prediction, compress local environments for phase identification, refine low-dimensional variables, and guide adaptive sampling. The role is deliberately limited: ML supplies representations and decision rules, while molecular physics still defines the systems, observables, and validation tests.

1.8.1 ML for Molecular Property Prediction

Supervised property prediction maps a molecular representation to a target such as solvation free energy, binding affinity, toxicity, or partition coefficient. Classical descriptors and fingerprints are efficient and interpretable, but they may lose three-dimensional or solvent-response information. Graph neural networks (GNNs) operate on molecular graphs and learn atom-level features through message passing [**Gilmer2017**]. Chapter 3 evaluates the Chemically Interpretable Graph Interaction Network (CIGIN), which models solute and solvent graphs jointly and represents their interaction rather than treating solvation as a single-molecule property [**Pathak2020**, **Pathak2021**]. It then extends the comparison to a trajectory-derived molecular representation learned from molecular dynamics frames and physical observables. The chapter asks whether good internal accuracy transfers across independent chemical datasets, and whether conformational features and applicability-domain descriptors reduce failures for molecules outside the training chemistry.

The solvation study is therefore the entry point for the thesis. A model cannot learn transferable physics if the representation omits the relevant physical factors, if the training data do not cover the chemical regimes where those factors matter, or if the model has no way to recognize unsupported chemotypes. Before ML can guide simulations, it must first pass a simpler test: whether its molecular representation still works when the chemical distribution changes and whether its limits can be made explicit.

1.8.2 ML for State and Phase Identification

In simulation analysis, ML can classify states or learn continuous structural maps from trajectory data. For condensed-phase systems, local atomic environments are often more informative than global averages. SOAP descriptors provide a symmetry-aware representation of local geometry by expanding neighbor densities in radial and angular basis functions [**Bartok2013**]. VAEs and related models can then compress these

descriptors into latent spaces that reveal low-dimensional structure.

Chapter 4 applies this idea to ice polymorph identification. The learned SOAP–VAE space functions as a data-driven order-parameter plane, resolving multiple ice phases and liquid-like environments under finite-temperature fluctuations. The aim is broader than assigning labels: the learned coordinates must also track phase evolution in simulations.

1.8.3 ML for Collective-Variable Discovery

For rare-event sampling, ML is most useful when it helps discover coordinates that preserve slow dynamics or mechanistic progress. TICA identifies linear combinations of features with maximal time-lagged autocorrelation [**PerezHernandez2013tICA**]. Autoencoders and VAEs learn nonlinear compressions of structural data [**Wehmeyer2018**, **Hernndez2018**]. Time-lagged autoencoders and VAMP-based methods incorporate dynamical objectives so that latent variables emphasize slow relaxation rather than only geometric variance [**Wu2020VAMP**]. Graph-neural-network variants of this variational approach extend slow-mode discovery to systems without a fixed atom ordering, learning symmetry-aware collective variables directly from molecular graphs [**Liu2023GraphVAMPnets**]. Diffusion maps approximate intrinsic manifold coordinates [**Rohrdanz2011**], and Boltzmann generators use normalizing flows to learn transformations between simple priors and molecular Boltzmann distributions [**Noe2019BoltzmannGenerators**].

These approaches do not make enhanced sampling automatic. Learned variables must still be validated against physical states, kinetics, and convergence behavior. In Chapter 5, ML appears in pathway clustering and neural path refinement, where noisy generated trajectories are converted into smoother pathway-aware CVs. The resulting PathCVs are useful because they preserve both progress along a transition and membership in a mechanistic route.

1.8.4 ML-Guided Adaptive Sampling

ML-guided adaptive sampling uses learned or evolving representations to decide where new simulations should begin. A typical loop begins with initial trajectories, projects the accumulated configurations into a physical or latent space, selects sparse, frontier, uncertain, or outlying regions, launches new simulations from selected configurations, and then updates the representation as additional data become available. The loop is useful precisely because the sampling coordinate can be imperfect at the beginning and still improve as the sampled ensemble becomes more informative.

Chapter 6 develops this logic in *TRAILS-MD*. The framework supports fixed CVs as well as TICA, VAE, TVAE, and related learned spaces. It applies density, Voronoi,

local-outlier, and farthest-point spawning rules, while recording parent–child trajectory lineage. That lineage is the key practical choice: it allows disconnected adaptive fragments to be reconstructed and assessed mechanistically. In this thesis, adaptive sampling is therefore treated as a representation-driven exploration strategy rather than as a direct replacement for rigorous kinetic estimators.

1.8.5 ML Interatomic Potentials and Foundation Models

A related, but separate, development is the use of ML to approximate potential energy surfaces. Neural network potentials, including Behler–Parrinello networks, Deep Potential, ANI, MACE, and NequIP, can approach quantum-mechanical accuracy at costs closer to classical simulation [**Behler2007**, **Behler2016**, **Zhang2018**, **Smith2017**, **Batatia2022**, **Batzner2022**]. These models preserve translational, rotational, and permutational symmetries and have enabled studies such as homogeneous ice nucleation with near-*ab initio* accuracy [**Piaggi2022**]. Broader molecular foundation models are also beginning to influence molecular representation learning and generative design. Large-scale self-supervised pretraining on SMILES corpora, exemplified by ChemBERTa, showed that transformer-style masked-language-modeling objectives can transfer to downstream molecular property prediction [**Chithrananda2020ChemBERTa**]. Uni-Mol extends this idea to 3D molecular representation learning by pretraining on large molecular-conformation and protein-pocket datasets for property prediction, conformation generation, and binding-pose estimation [**Zhou2023UniMol**]. Recent reviews have begun to organize the resulting landscape of unimodal and multimodal molecular foundation models and their pretraining strategies [**Song2026MolecularFoundationReview**]. These methods are included for context rather than as thesis contributions: the work here uses ML mainly for representations, classification, and sampling decisions, not for learning new interatomic potentials or foundation models.

1.9 Toward Automated Molecular Discovery Workflows

The larger motivation is a closed-loop simulation workflow. Such a workflow would propose molecular candidates or starting configurations, predict their properties, simulate their structural and dynamical behavior, identify relevant states and transition pathways, estimate thermodynamic or kinetic observables, and use the resulting information to refine the next round of predictions or simulations. Experimental autonomous-discovery platforms already follow a related design–make–test–analyze logic guided by machine-learned property models [**Koscher2023AutonomousDiscovery**], and self-driving laboratories extend the idea across chemistry and materials science [**Tom2024SelfDrivingLabs**]. The present thesis develops the simulation-side pieces of that loop: representation, identification, and sampling.

The four application chapters contribute different pieces of this loop: transfer across chemical spaces, local phase identification in water, rapid path generation connected to kinetic estimation, and lineage-aware adaptive sampling for conformational exploration.

Their common dependency is representation, but the requirement changes across the thesis. A solvation model needs encodings that transfer across molecules and expose when a molecule lies outside the supported chemical domain. Ice classification needs coordinates that distinguish local environments under thermal fluctuations. Pathway generation needs coordinates that preserve both progress and route identity. Adaptive sampling needs latent spaces that can be updated as new data arrive. The dissertation therefore moves from static and trajectory-derived molecular encodings, to learned structural order parameters, to path and latent spaces used for rare-event sampling.

1.10 Thesis Scope and Chapter Roadmap

The remainder of the thesis develops this progression in four application chapters.

Chapter 2 provides the theoretical and methodological foundation for the remainder of the thesis, including molecular dynamics, force fields, molecular representations, dimensionality reduction, supervised learning, rare-event sampling, Weighted Ensemble simulations, and Markov state modeling.

Chapter 3: Transferability of Molecular Representations for Aqueous Solvation Free-Energy Prediction. In Chapter 3 we study the aqueous solvation free energy, a thermodynamic property that is important for solubility, partitioning, and molecular design. The chapter compares physicochemical descriptors, molecular fingerprints, graph-based representations, and trajectory-derived learned embeddings across MNSol, FreeSolv and CombiSolv [**Mobley2014**, **Vermeire2021**, **YuSolvbert2023**]. Its central test is transferability: whether a model trained on one chemical distribution is still predictive on another, and how that behavior depends on representation, training-set diversity, the physical components of solvation, and explicit applicability-domain information. The final trajectory-informed analysis shows that conformational and physical proxies improve some external predictions, but that rare chemotypes, shielded polarity, ionizable neutral forms, and unsupported elements must be exposed to the model rather than hidden inside a generic molecular vector.

Chapter 4: IceCoder: Self-Supervised Identification of Ice Polymorphs. Chapter 4 shifts from molecular property prediction to structural identification in condensed-phase water. It introduces *IceCoder*, a self-supervised SOAP–VAE framework that maps local oxygen environments into a two-dimensional latent representation. This learned space separates multiple ice polymorphs and liquid water, supports support-vector-

machine phase assignment, and tracks phase evolution in two-phase and three-phase simulations. The chapter replaces a fixed hand-crafted order parameter with a learned, physically validated latent order-parameter space.

Chapter 5: Pathway Generation and Path-Resolved Kinetics. Chapter 5 studies rare events and multi-pathway transitions through *PathGennie*, which generates transition pathways by direction-guided selection of short unbiased MD segments. The path-resolved kinetics workflow then clusters generated paths using DTW, refines them into smooth reference curves using neural networks, converts them into PathCVs, and uses them to run pre-seeded WE simulations. This framework connects exploratory pathway discovery with quantitative route-specific rate constants for model potentials, ligand unbinding, and protein conformational transitions.

Chapter 6: Lineage-Aware Adaptive Sampling. Chapter 6 presents *TRAILS-MD*, a lineage-aware adaptive sampling framework for conformational exploration. Instead of targeting a single predefined transition path, *TRAILS-MD* iteratively propagates parallel walkers, projects configurations into fixed or learned sampling spaces such as TICA and TVAE coordinates, and spawns new simulations from sparse or frontier regions using density, Voronoi, local-outlier, or farthest-point criteria. Explicit parent–child lineage tracking allows discontinuous adaptive fragments to be reconstructed into mechanistic pathways and used as starting points for downstream kinetic analysis, providing a practical response to the circular coordinate-discovery problem introduced earlier.

1.11 Objectives and Contributions

The objective of this thesis is to develop and evaluate machine-learning-assisted workflows for molecular prediction, structural identification, and rare-event sampling. The specific contributions are:

1. A transferability-focused evaluation of descriptor, fingerprint, graph, and trajectory-derived representations for aqueous solvation free-energy prediction across independent chemical datasets, including explicit analysis of applicability-domain failures.

2. The development of *IceCoder*, a self-supervised SOAP–VAE framework for mapping and identifying local ice environments through a continuous latent order-parameter space.

3. The design of *PathGennie*, a direction-guided pathway-generation algorithm that uses short unbiased MD segments to discover rare-event transition routes without modifying the Hamiltonian.

4. A path-resolved kinetic workflow that combines DTW clustering, neural path refinement, PathCV construction, and WE simulations to estimate route-specific rate constants.

5. The implementation of *TRAILS-MD*, a restartable, lineage-aware adaptive sampling framework that couples fixed or learned representations with trajectory spawning for automated conformational exploration.

Taken together, these contributions treat representation learning as useful only when it is embedded in a physically constrained workflow: a predictor must transfer, a latent space must track real structural change, and a sampling coordinate must lead to interpretable pathways or kinetics. The next chapter establishes the theoretical and methodological background required for those tests.

2

Theoretical Background and Methodology

The introduction framed this dissertation as a pipeline from molecular coordinates to representations, models, states or collective variables (CVs), sampling, and discovery. This chapter supplies the technical pieces needed to follow that pipeline through the four results chapters. It is not a general review of molecular dynamics (MD) or machine learning (ML); it focuses on the methods that determine what is represented, how states or pathways are identified, and how rare events are sampled.

The *representation bottleneck* begins with raw Cartesian coordinates: they must be converted into compact, invariant, and informative descriptors without losing the physicochemical features that control the phenomenon of interest. The *identification bottleneck* appears after featurization, when descriptors must be converted into discrete states, phases, pathways, or progress coordinates. The *sampling bottleneck* enters when those representations and states must be supported by enough configurational and dynamical data to estimate thermodynamic or kinetic quantities for rare-event processes. Each bottleneck recurs across multiple results chapters, and the methods described below are selected accordingly.

The chapter starts with the MD foundations and force-field assumptions used later, with the broader statistical-mechanical motivation left to Chapter 1. It then moves through the representations used for solvation prediction, local ice-phase identification, and transition-path analysis, followed by dimensionality reduction, latent-space learning, supervised prediction and classification, path-based rare-event methods, and Markov state modeling. These sections define the technical vocabulary used in Chapters 3–6.

2.1 Molecular Dynamics Foundations

The statistical-mechanical role of MD and the finite-time sampling problem were introduced in Sections 1.2 and 1.4. Here, only the operational ingredients required by the later methodology are recalled. For a system of N interacting particles, classical MD propagates positions by numerically integrating Newton's second law,

$$m_i \frac{d^2\mathbf{r}_i}{dt^2} = \mathbf{F}_i = -\nabla_i V(\mathbf{r}_1, \ldots, \mathbf{r}_N), \tag{2.1}$$

where m_i and $\mathbf{r}_i$ are the mass and position of particle i, and V is the potential energy surface (PES). In this dissertation, MD trajectories provide the molecular coordinates that are subsequently converted into descriptors, fingerprints, graphs, SOAP vectors, CVs, latent spaces, or transition pathways.

2.1.1 Verlet and Velocity Verlet Integrators

The numerical integrator matters because all downstream analysis assumes that the generated trajectories are physically meaningful discretizations of the underlying dynamics. For this reason, MD integrators are usually chosen to be time-reversible and symplectic, limiting long-time energy drift and preserving the qualitative structure of phase-space evolution.

The original Verlet algorithm follows from Taylor expansions of the position forward and backward in time:

$$\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{F}_i(t)}{2m_i}\Delta t^2 + \frac{\dddot{\mathbf{r}}_i(t)}{6}\Delta t^3 + O(\Delta t^4), \tag{2.2}$$

$$\mathbf{r}_i(t-\Delta t) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\Delta t + \frac{\mathbf{F}_i(t)}{2m_i}\Delta t^2 - \frac{\dddot{\mathbf{r}}_i(t)}{6}\Delta t^3 + O(\Delta t^4). \tag{2.3}$$

Adding these equations yields

$$\mathbf{r}_i(t+\Delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\Delta t) + \frac{\mathbf{F}_i(t)}{m_i}\Delta t^2 + O(\Delta t^4). \tag{2.4}$$

This update is simple and stable, but it does not propagate velocities explicitly. Modern MD engines therefore commonly use the Velocity Verlet form,

$$\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\mathbf{F}_i(t)}{2m_i}\Delta t^2, \tag{2.5}$$

$$\mathbf{v}_i\left(t+\frac{\Delta t}{2}\right) = \mathbf{v}_i(t) + \frac{\mathbf{F}_i(t)}{2m_i}\Delta t, \tag{2.6}$$

$$\mathbf{F}_i(t+\Delta t) = -\nabla_i V(\mathbf{r}(t+\Delta t)), \tag{2.7}$$

$$\mathbf{v}_i(t+\Delta t) = \mathbf{v}_i\left(t+\frac{\Delta t}{2}\right) + \frac{\mathbf{F}_i(t+\Delta t)}{2m_i}\Delta t. \tag{2.8}$$

The same integration principle underlies the unbiased trajectory propagation used in the WE calculations in Chapter 5 and the short walker trajectories used for exploratory adaptive sampling in Chapter 6.

2.1.2 Leapfrog Integrator

A closely related variant is the leapfrog integrator, which updates positions and velocities on interleaved half-integer time steps:

$$\mathbf{v}_i\left(t + \frac{\Delta t}{2}\right) = \mathbf{v}_i\left(t - \frac{\Delta t}{2}\right) + \frac{\mathbf{F}_i(t)}{m_i}\Delta t, \tag{2.9}$$

$$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i\left(t + \frac{\Delta t}{2}\right)\Delta t. \tag{2.10}$$

The velocities at integer times can be recovered by averaging,

$$\mathbf{v}_i(t) = \frac{1}{2}\left[\mathbf{v}_i\left(t - \frac{\Delta t}{2}\right) + \mathbf{v}_i\left(t + \frac{\Delta t}{2}\right)\right]. \tag{2.11}$$

Like Velocity Verlet, the leapfrog scheme is symplectic and time-reversible, and it introduces the same global $O(\Delta t^2)$ error in positions. The two algorithms are algebraically equivalent: a shift of the velocity array by half a timestep transforms one into the other. The leapfrog formulation is used by several major MD packages and appears in the sampling literature because its half-step velocity storage is convenient for coupling to thermostats, including the Langevin thermostat described next.

2.1.3 Langevin Dynamics

The Newtonian dynamics in Eq. (2.1) preserve energy and sample the microcanonical (NVE) ensemble. Many of the applications in this thesis require canonical (NVT) sampling at controlled temperature, or benefit from the enhanced exploration that a stochastic thermostat provides. Langevin dynamics introduces a friction term and a random force into the equation of motion,

$$m_i \frac{d^2\mathbf{r}_i}{dt^2} = \mathbf{F}_i - \gamma_i m_i \frac{d\mathbf{r}_i}{dt} + \mathbf{R}_i(t), \tag{2.12}$$

where γ_i is a friction coefficient (with units of inverse time) and $\mathbf{R}_i(t)$ is a Gaussian white-noise process satisfying the fluctuation–dissipation theorem,

$$\langle \mathbf{R}_i(t) \rangle = 0, \tag{2.13}$$

$$\langle \mathbf{R}_i(t)\, \mathbf{R}_j(t') \rangle = 2\gamma_i m_i k_B T\, \delta_{ij}\delta(t - t')\, \mathbf{I}. \tag{2.14}$$

The noise magnitude is set so that the injected thermal energy exactly balances the energy dissipated by friction, driving the system toward a Boltzmann distribution at temperature T.

In the limit $\gamma_i \to 0$, the Langevin equation reduces to Newtonian dynamics, and the

scheme can be used as a pure thermostat without significantly perturbing dynamical properties. For the WE calculations in Chapter 5, Langevin dynamics provides the stochastic propagation that generates trajectory diversity within each bin. For the adaptive-sampling loops in Chapter 6, it supplies the thermostat that maintains constant temperature during the short exploration trajectories.

A common discrete-time integration scheme for Langevin dynamics is the BAOAB splitting, which alternates drift, noise injection, and friction steps to preserve the correct configurational distribution while retaining second-order accuracy in the deterministic limit. In that scheme, the half-step velocity storage of the leapfrog formulation (Section 2.1.2) becomes natural because the noise term can be applied at the half-step point. Throughout this thesis, all Langevin integration is performed using the BAOAB or a closely related Verlet-type splitting, ensuring that the NVT ensemble is sampled correctly even when the friction coefficient is large enough to accelerate barrier crossing.

2.2 Potential Energy Surfaces and Force Fields

The reliability of any MD-derived representation or kinetic estimate depends on the PES used to generate the underlying coordinates. For the system sizes and timescales considered here, this PES is represented by empirical classical force fields rather than by solving the electronic-structure problem at each time step.

2.2.1 The Born–Oppenheimer Approximation

The hierarchy of molecular models was discussed in Section 1.3. At the quantum-mechanical level, the PES is obtained by solving the electronic Schrodinger equation at fixed nuclear coordinates. The Born–Oppenheimer approximation separates the fast electronic response from the slower nuclear motion, giving

$$\hat{H}_{\text{elec}}\Psi_{\text{elec}}(\mathbf{r};\mathbf{R}) = E_{\text{elec}}(\mathbf{R})\Psi_{\text{elec}}(\mathbf{r};\mathbf{R}). \tag{2.15}$$

The electronic energy, together with nuclear–nuclear repulsion, defines the PES. Classical MD replaces this surface with parameterized bonded and non-bonded interaction functions. That replacement is what makes the sampling-heavy parts of the thesis feasible, because those chapters require large ensembles of trajectories rather than isolated high-accuracy energy evaluations.

2.2.2 Functional Form of Empirical Force Fields

Empirical force fields represent the PES as a sum of bonded and non-bonded terms:

$$V(\mathbf{R}) = V_{\text{bond}} + V_{\text{angle}} + V_{\text{dihed}} + V_{\text{vdW}} + V_{\text{elec}}. \tag{2.16}$$

Bonded Interactions

Bonded terms describe local covalent geometry:

1. **Bond stretching:**
$$V_{\text{bond}} = \sum_{\text{bonds}} \frac{1}{2} k_b (b - b_0)^2, \tag{2.17}$$
where b_0 is the equilibrium bond length.

2. **Angle bending:**
$$V_{\text{angle}} = \sum_{\text{angles}} \frac{1}{2} k_\theta (\theta - \theta_0)^2, \tag{2.18}$$
where θ_0 is the equilibrium bond angle.

3. **Torsional terms:**
$$V_{\text{dihed}} = \sum_{\text{torsions}} \sum_{n} \frac{1}{2} V_n \left[1 + \cos(n\phi - \gamma)\right], \tag{2.19}$$
where ϕ is the dihedral angle and γ is the phase shift.

Non-Bonded Interactions

Non-bonded terms account for steric, dispersive, and electrostatic interactions:

1. **van der Waals interactions:**
$$V_{\text{vdW}} = \sum_{i<j} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right]. \tag{2.20}$$
The repulsive term approximates excluded-volume effects, while the attractive term models dispersion.

2. **Electrostatic interactions:**
$$V_{\text{elec}} = \sum_{i<j} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}. \tag{2.21}$$

These terms determine the trajectories from which the thesis derives molecular representations, phase assignments, pathways, and kinetic estimates. Any force-field limitation is therefore inherited by the learned descriptors, phase labels, and sampling calculations built on those trajectories.

A force field is not a unique approximation to the molecular PES. Different parameter sets are optimized against different combinations of quantum-mechanical calculations,

condensed-phase thermodynamic properties, and structural or spectroscopic measurements. Consequently, the appropriate model depends on the chemical composition of the system and on the physical observables of interest.

For biomolecular simulations, widely used protein force-field families include AMBER, CHARMM, and OPLS. The AMBER force fields employ atom-centred fixed partial charges together with harmonic bonded terms, Lennard–Jones interactions, and periodic torsional potentials. Successive protein parameter sets have refined the backbone and side-chain torsional parameters; for example, ff14SB was developed to improve the description of protein secondary structure and side-chain conformational preferences relative to earlier AMBER parameter sets [**Maier2015**]. CHARMM36m similarly refines the CHARMM36 protein potential, with particular attention to the conformational balance of folded proteins, peptides, and intrinsically disordered regions [**Huang2016**]. The OPLS-AA family provides another all-atom framework for proteins, peptides, and organic molecules, with parameters fitted primarily to reproduce molecular conformational energetics and condensed-phase properties [**Jorgensen1996**].

Small organic molecules and ligands often require parameter sets with broader chemical coverage than protein-specific force fields. The General AMBER Force Field (GAFF), for example, extends the AMBER functional form to a wide range of organic functional groups and is frequently combined with AMBER protein force fields in simulations of protein–ligand complexes and solvated small molecules [**Wang2004**]. Comparable general-purpose parameter sets include the CHARMM General Force Field and extensions of the OPLS family. Although these force fields share broadly similar functional forms, their atom types, partial charges, torsional parameters, and combination rules are not generally interchangeable.

Solvent models constitute an additional component of the overall force-field description. Common fixed-charge water models include the three-site TIP3P and SPC/E models and the four-site TIP4P family [**Jorgensen1983**, **Berendsen1987**]. In three-site models such as TIP3P, the interaction sites coincide with the oxygen and hydrogen nuclei, with the Lennard–Jones interaction normally assigned to oxygen. Four-site models displace the negative charge from the oxygen nucleus to a massless site located along the H–O–H bisector. This shifted charge distribution generally provides a more directional representation of water electrostatics while retaining a computationally inexpensive rigid geometry.

Different members of the TIP4P family were parameterized for different physical regimes. TIP4P/2005 was developed as a general-purpose model for the condensed phases of water and provides an improved description of several liquid and solid-state properties compared with the original TIP4P parameterization [**Abascal2005TIP4P2005**]. TIP4P/Ice was instead optimized specifically to reproduce the melting behavior and relative stability of crystalline ice phases [**Abascal2005**]. The latter model was therefore

used for the ice-polymorphism calculations in Chapter 4. In that application, the objective was to learn representations that distinguish liquid water, hexagonal ice, cubic ice, and related local environments. A water model designed for crystalline phase behaviour was consequently more appropriate than a model optimized primarily for ambient liquid water.

The choice of force field and solvent model must therefore be considered part of the physical definition of a simulation rather than a purely technical setting. A protein force field controls the balance among secondary structures and side-chain conformations; a ligand force field determines intramolecular energetics and intermolecular contacts; and the water model affects solvation structure, hydrogen-bond dynamics, diffusion, and phase stability. The trajectories analysed in this thesis should accordingly be interpreted as samples from the PES defined by the selected combination of solute and solvent parameters. Learned representations, state assignments, transition pathways, and kinetic estimates inherit both the strengths and the limitations of that underlying model.

2.3 Molecular Representations

Raw Cartesian coordinates are complete but inconvenient: they are high-dimensional, frame-dependent, and not automatically invariant to translation, rotation, or permutation of equivalent atoms. Each results chapter therefore begins by choosing a representation suited to the scientific task, whether that task is solvation prediction, ice-phase identification, pathway analysis, or adaptive sampling.

2.3.1 Coordinate Invariances

Let a molecular configuration be represented by $\mathbf{R} \in \mathbb{R}^{3N}$. A model $f(\mathbf{R})$ predicting a property, local phase, or progress coordinate should respect the physical symmetries of the system:

1. **Translation:** $f(\mathbf{R} + \mathbf{t}) = f(\mathbf{R})$ for any vector $\mathbf{t} \in \mathbb{R}^3$.
2. **Rotation:** $f(\mathbf{UR}) = f(\mathbf{R})$ for any proper rotation matrix $\mathbf{U} \in SO(3)$.
3. **Permutation:** $f(\mathbf{PR}) = f(\mathbf{R})$ for any permutation matrix $\mathbf{P}$ swapping identical atoms.

The representation choices below enforce or approximate these invariances in different ways.

2.3.2 Descriptor-Based Molecular Representations

For solvation free-energy prediction in Chapter 3, small molecules are represented using descriptor vectors computed from molecular structure. Descriptor-based modeling converts a compound into features such as topological polar surface area (TPSA), hydrogen-bond donor and acceptor counts, molecular weight, functional-group counts, quantitative estimate of drug-likeness (QED), and octanol–water partition coefficient (logP). These features are attractive because they are inexpensive and chemically interpretable.

A hand-crafted descriptor set can omit structural motifs or intermolecular effects that control solvation. Descriptor models are useful here because they make that limitation explicit: they test how much transferable solvation information can be captured by global physicochemical features alone.

2.3.3 Molecular Fingerprints

Molecular fingerprints encode molecular substructures as fixed-length bit vectors. They are used in Chapter 3 to test whether local structural patterns improve transferability relative to global descriptors. Structural keys use predefined fragment dictionaries. MACCS keys are widely used structural keys [**DurantReoptimization2002**], and the public 166-key subset is used in this work. PubChem fingerprints are 881-bit structural keys used for similarity searches and neighboring tasks [**FernndezdeGortari2017**].

Hashed fingerprints generate molecular fragments dynamically and map them into a fixed-length vector using a hash function. This increases flexibility but can introduce bit collisions. Fingerprints in this thesis are Atom-Pair (AP) fingerprints [**CarhartAtom1985**], RDKit fingerprints [**rdkit**], Morgan fingerprints [**Morgan1965**] and Extended Connectivity Fingerprints (ECFPs) [**Rogers2010**]. Fingerprint length controls the trade-off between structural resolution and sparsity; in the solvation study, 2048-bit fingerprints provide a practical balance between information content and model performance. An overview of the molecular representations used in this work is shown in Fig. 2.1.

2.3.4 Graph-Based Molecular Representations

Graph representations address the same solvation prediction problem from a different inductive bias. A molecular graph $G(V, E)$ represents atoms as nodes and bonds as edges. Each node $i \in V$ carries features such as atomic number, atom type, formal charge, hybridization, aromaticity, and valence. Bonds define the edge set E and the adjacency matrix

$$A_{ij} = \begin{cases} 1, & \text{if atoms } i \text{ and } j \text{ are bonded,} \\ 0, & \text{otherwise.} \end{cases} \tag{2.22}$$

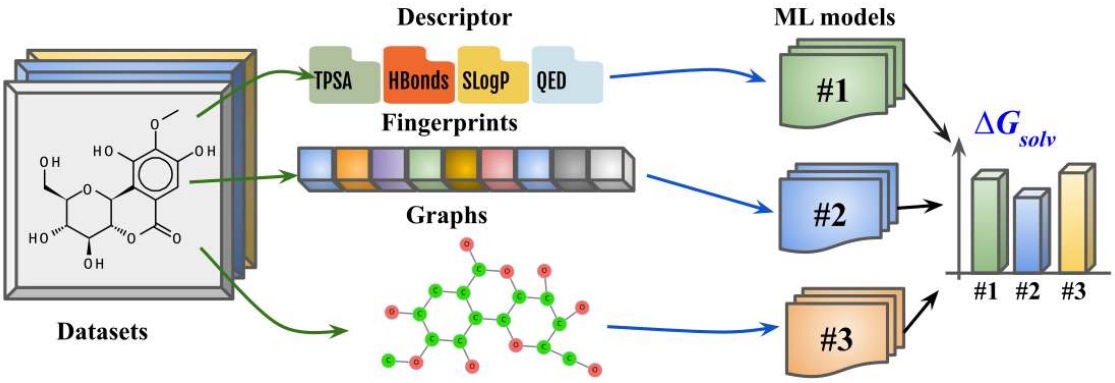

Figure 2.1: Schematic overview of molecular representations. From SMILES strings, descriptor-based and fingerprint-based representations are generated for classical machine learning models. For graph-based models, molecular graphs are constructed with node features and adjacency matrices and fed to different architectures for mapping to the target property.

Edges may encode bond order, conjugation, ring membership, and stereochemistry.

This representation is used in Chapter 3 by the Chemically Interpretable Graph Interaction Network (CIGIN), where solute and solvent graphs are processed jointly. Graphs are less manually constrained than descriptors or fingerprints and allow the model to learn atom-level interaction patterns, but they require careful validation because added flexibility can reduce transferability when training data are limited.

2.3.5 Collective Variables in Molecular Dynamics

The identification bottleneck appears when high-dimensional configurations must be converted into states, phases, pathways, or progress coordinates. A collective variable (CV) is a low-dimensional function of atomic coordinates,

$$\xi(\mathbf{R}) = [\xi_1(\mathbf{R}), \xi_2(\mathbf{R}), \ldots, \xi_d(\mathbf{R})], \qquad d \ll 3N, \tag{2.23}$$

where $\mathbf{R}$ denotes the full molecular configuration. A useful CV should distinguish metastable states, resolve transition pathways, and provide a meaningful coordinate for free-energy analysis, binning, or adaptive sampling. An example of such a free-energy projection is shown in Fig. 2.2.

Physics-based CVs, such as distances, dihedrals, root-mean-square deviation (RMSD), radius of gyration, contact fractions, coordination numbers, hydrogen-bond counts, and orientational order parameters, are interpretable and inexpensive. They are used throughout the thesis as baselines, analysis coordinates, and fixed sampling spaces. Their weakness is degeneracy: a small number of scalar CVs can collapse distinct mechanisms onto the same value, especially in multi-pathway rare events. The learned latent spaces in Chapter 4, the path collective variables (PathCVs) in Chapter 5, and the

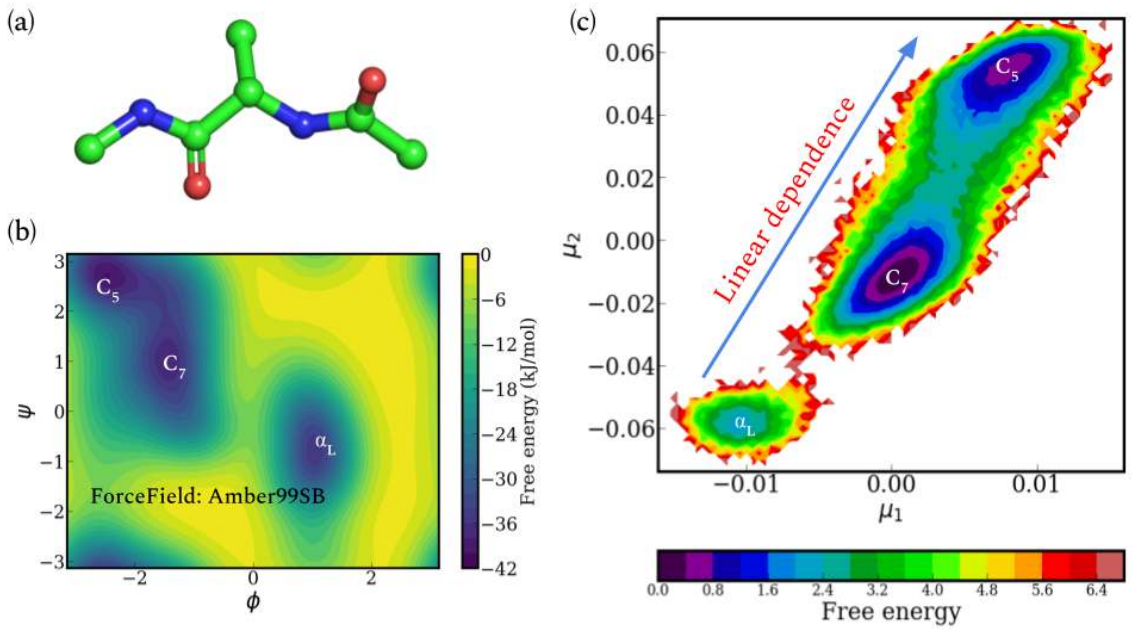

Figure 2.2: Free energy projection of the alanine dipeptide system in low-dimensional CV space. (a) Alanine dipeptide is shown in ball-and-stick model. (b) Free energy surface projected onto the backbone dihedral angles ϕ and ψ. (c) A learned representation of the system in the latent space of a variational autoencoder.

adaptive projection models in Chapter 6 are all responses to that problem.

2.3.6 Classical Order Parameters for Water and Ice

For water and ice systems, classical order parameters quantify local structure, hydrogen-bond organization, packing geometry, and orientational symmetry. They provide interpretable baselines for Chapter 4, where learned SOAP–VAE coordinates are compared against hand-crafted descriptors.

Tetrahedral Order Parameter

The tetrahedral order parameter Q quantifies deviations from ideal tetrahedral coordination [**Chau1998**, **Errington2001**]. For a central water molecule i,

$$Q(i) = 1 - \frac{3}{8} \sum_{j=1}^{3} \sum_{k=j+1}^{4} \left(\cos \phi_{jk} + \frac{1}{3} \right)^2 , \tag{2.24}$$

where ϕ_{jk} is the angle formed by the central molecule and its four nearest oxygen neighbors. An ideal tetrahedral environment gives $Q = 1$. The parameter is intuitive but contains limited information about longer-range ordering and polymorph-specific motifs.

Local Structure Index

The local structure index (LSI) measures the separation between the first and second coordination shells [**Shiratani1996**, **Shiratani1998**, **Wikfeldt2011**]. Neighbor distances are sorted as

$$r_1 < r_2 < \cdots < r_n < r_c < r_{n+1}, \tag{2.25}$$

where r_c is a cutoff. With

$$\Delta_i = r_{i+1} - r_i, \tag{2.26}$$

the LSI is

$$I = \frac{1}{n} \sum_{i=1}^{n} (\Delta_i - \bar{\Delta})^2. \tag{2.27}$$

Large values indicate a well-separated first shell, while small values indicate more densely packed or disordered environments. Because LSI is radial, it may miss orientational distinctions between ice polymorphs.

Four-Body Orientational Order Parameters

The F_4 order parameter includes orientational correlations through the water-dimer dihedral angle ϕ [**Rodger1996**, **Jimnezngeles2014**, **Choudhary2018**, **Choudhary2020**]:

$$F_4 = \langle \cos(3\phi) \rangle. \tag{2.28}$$

It is sensitive to cage-like structures and complements radial measures, but as a single averaged quantity it cannot uniquely identify all complex polymorphic environments.

Voronoi-Based Descriptors

Voronoi tessellation partitions space into cells surrounding each molecule, enabling local packing and free-volume measures without a simple radial cutoff [**BERNAL1959**, **Stirnemann2012**]. The Voronoi volume V_i gives a local density estimate,

$$\rho_i = \frac{1}{V_i}, \tag{2.29}$$

and the cell asphericity,

$$\eta_i = \frac{A_i^3}{36\pi V_i^2}, \tag{2.30}$$

measures deviations from spherical symmetry, where A_i is the cell surface area [**Ruocco1992**]. These descriptors are useful for packing and interfacial environments but contain little explicit hydrogen-bond orientation information.

Steinhardt Bond-Orientational Order Parameters

Steinhardt order parameters characterize local rotational symmetry through spherical harmonics [**Steinhardt1983**, **Lechner2008**, **Reinhardt2012**]. For particle i,

$$q_l(i) = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |q_{lm}(i)|^2}, \tag{2.31}$$

where

$$q_{lm}(i) = \frac{1}{N_b(i)} \sum_{j=1}^{N_b(i)} Y_{lm}(\hat{\mathbf{r}}_{ij}). \tag{2.32}$$

Different values of l probe different local symmetries. Locally averaged variants reduce thermal noise, but performance depends on neighbor definitions and phase distributions can still overlap [**Mickel2013**].

Entropy-Based Fingerprints

Entropy-based descriptors connect local structure to pair correlations. The two-body excess entropy is [**Nettleton1958**, **Baranyai1989**, **Truskett2000**, **Saija2003**]

$$S_2 = -2\pi\rho k_B \int_0^\infty \left[g(r) \ln g(r) - g(r) + 1 \right] r^2 dr. \tag{2.33}$$

Local entropy fingerprints replace the global radial distribution function with a local average [**Piaggi2017**]. They provide physically meaningful order measures but primarily reflect radial correlations, motivating richer invariant representations for polymorph classification.

2.3.7 Smooth Overlap of Atomic Positions (SOAP)

Smooth Overlap of Atomic Positions (SOAP) descriptors overcome the limitations of scalar order parameters by encoding the full local neighbor density around each atom and building rotationally invariant features from that density [**De2016**, **Maksimov2020**, **Himanen2020**]. This makes SOAP suitable for Chapter 4, where subtle ice-polymorph distinctions depend on packing, angular order, and many-body local correlations.

The local neighbor density $\rho_i(\mathbf{r})$ around a central atom i is represented by placing Gaussians on its neighbors j:

$$\rho_i(\mathbf{r}) = \sum_j \exp\left[-\frac{|\mathbf{r} - \mathbf{r}_{ij}|^2}{2\sigma^2} \right]. \tag{2.34}$$

This density is expanded in radial functions $g_n(r)$ and spherical harmonics $Y_{lm}(\theta, \phi)$:

$$\rho_i(\mathbf{r}) = \sum_{nlm} c^{(i)}_{nlm} g_n(r) Y_{lm}(\theta, \phi), \tag{2.35}$$

with coefficients

$$c^{(i)}_{nlm} = \int \rho_i(\mathbf{r}) g_n(r) Y^*_{lm}(\theta, \phi)\, d\mathbf{r}. \tag{2.36}$$

Rotational invariance is obtained by forming the power spectrum,

$$p^{(i)}_{nn'l} = \sqrt{\frac{8\pi^2}{2l+1}} \sum_m \left(c^{(i)}_{nlm}\right)^* c^{(i)}_{n'lm}. \tag{2.37}$$

The vector $\mathbf{p}^{(i)} = \{p^{(i)}_{nn'l}\}$ is invariant to translation, rotation, and permutation of equivalent atoms. In Chapter 4, SOAP vectors serve as the high-dimensional representation from which VAE latent coordinates and phase assignments are derived.

2.4 Dimensionality Reduction and Latent Variables

High-dimensional descriptors preserve structural information, but most downstream tasks require a smaller coordinate system. Ice phases must be visualized and classified, WE simulations need bins, and adaptive sampling needs spaces from which new starting points can be selected. Dimensionality reduction maps a descriptor vector $\mathbf{x}_i \in \mathbb{R}^D$ to a latent coordinate

$$\mathbf{z}_i = f(\mathbf{x}_i), \qquad \mathbf{z}_i \in \mathbb{R}^d, \quad d \ll D. \tag{2.38}$$

In this thesis, such latent spaces are used as order-parameter planes for ice phases, as learned sampling spaces for adaptive MD, and as compact coordinates for comparing molecular configurations.

2.4.1 Principal Component Analysis

Principal component analysis (PCA) provides a deterministic linear baseline [**Pearson1901**]. After centering the data, PCA constructs the covariance matrix

$$\mathbf{C} = \frac{1}{M-1} \sum_{i=1}^{M} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T, \tag{2.39}$$

where M is the number of samples and $\bar{\mathbf{x}}$ is the mean descriptor vector. The principal components satisfy

$$\mathbf{C}\mathbf{w}_k = \lambda_k \mathbf{w}_k, \tag{2.40}$$

and the projection is

$$z_{ik} = \mathbf{w}_k^T(\mathbf{x}_i - \bar{\mathbf{x}}), \qquad k = 1, \ldots, d. \tag{2.41}$$

PCA is used as an interpretable compression method and baseline. Its limitation is that it emphasizes variance rather than slow dynamics or non-linear state separation, which can be problematic when metastable states overlap in linear projections.

2.4.2 Autoencoders

An autoencoder (AE) is a non-linear dimensionality reduction model. The encoder maps the input descriptor to a latent vector,

$$\mathbf{z} = f_\theta(\mathbf{x}), \tag{2.42}$$

and the decoder reconstructs the descriptor,

$$\hat{\mathbf{x}} = g_\psi(\mathbf{z}) = g_\psi(f_\theta(\mathbf{x})). \tag{2.43}$$

Training minimizes a reconstruction loss such as

$$\mathcal{L}_{\mathrm{AE}} = \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 . \tag{2.44}$$

AEs can learn curved manifolds, but a standard AE does not impose a probabilistic structure on the latent space. This can make the latent coordinates less smooth or less convenient for sampling. Figure 2.3 places this deterministic compression alongside PCA and the probabilistic VAE formulation used later.

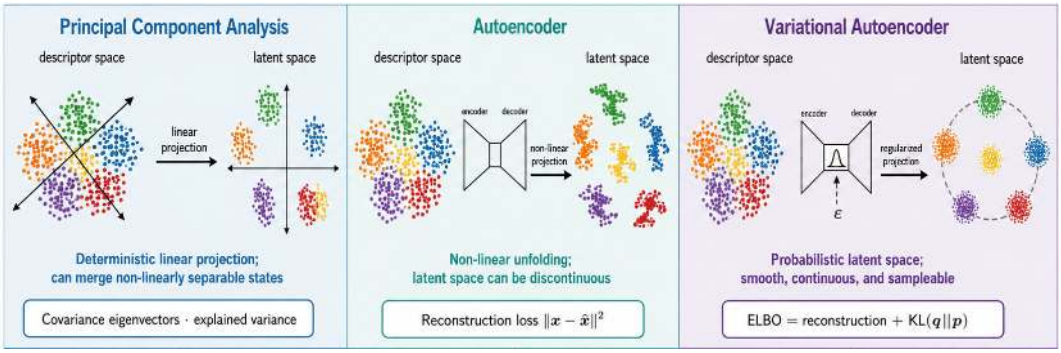

Figure 2.3: Comparative schematic of descriptor compression using PCA, autoencoders, and variational autoencoders.

2.4.3 Variational Autoencoders

A variational autoencoder (VAE) regularizes the latent space by treating the latent coordinate as a probability distribution [**VAE**, **Hernndez2018**]. The encoder predicts

an approximate posterior,

$$q_\theta(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\boldsymbol{\mu}_\theta(\mathbf{x}), \mathrm{diag}(\boldsymbol{\sigma}^2_\theta(\mathbf{x}))\right), \tag{2.45}$$

from which latent points are sampled using

$$\mathbf{z} = \boldsymbol{\mu}_\theta(\mathbf{x}) + \boldsymbol{\sigma}_\theta(\mathbf{x}) \odot \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}). \tag{2.46}$$

The model minimizes the negative evidence lower bound,

$$\mathcal{L}_{\mathrm{VAE}} = -\mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}\left[\log p_\psi(\mathbf{x}|\mathbf{z})\right] + D_{\mathrm{KL}}\left(q_\theta(\mathbf{z}|\mathbf{x}) \,\|\, p(\mathbf{z})\right), \tag{2.47}$$

where $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$ is usually a standard normal prior. In Chapter 4, the VAE provides a non-linear latent order-parameter space for SOAP descriptors, allowing phase boundaries to be learned in a compact representation. The VAE workflow is illustrated in Fig. 2.4.

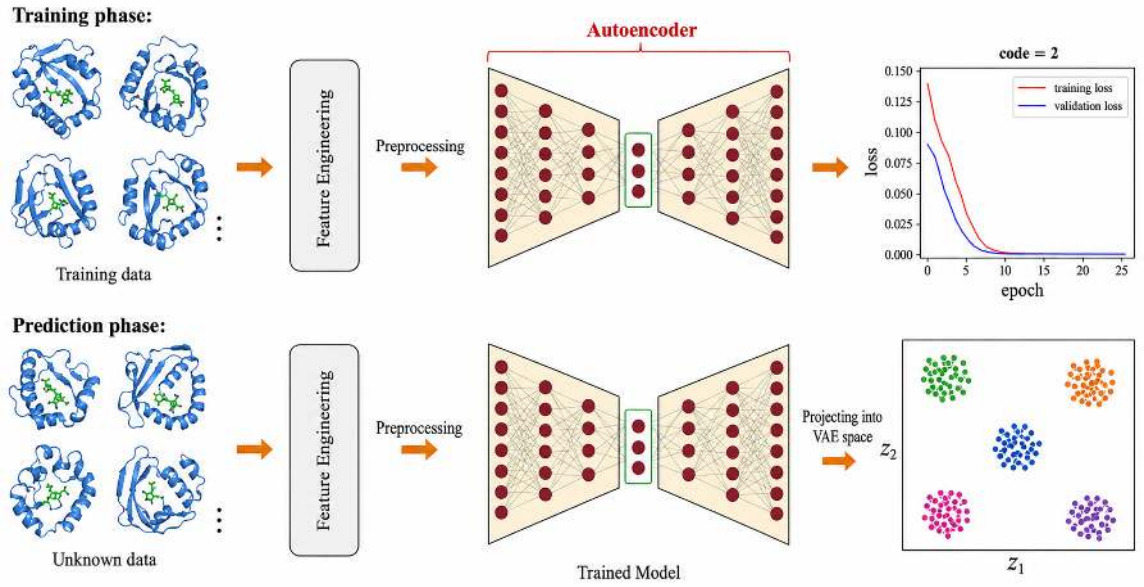

Figure 2.4: Schematic overview of the VAE workflow. The system is featurized using a suitable invariant representation and passed to the encoder network to produce a low-dimensional latent coordinate. The decoder reconstructs the original input from the latent coordinate, and the model is trained by minimizing the negative evidence lower bound. The figure also illustrates a representative training-loss curve and the projection of data into the learned latent space.

2.4.4 Time-Lagged Projection Models

Chapter 6 uses adaptive sampling spaces that should capture slow conformational changes rather than only high-variance motions. Time-lagged independent component

analysis (TICA) seeks linear combinations of features that are maximally autocorrelated over a lag time τ [**No2013**, **PerezHernandez2013tICA**]. It solves

$$\mathbf{C}_{0\tau}\,\mathbf{w}_i = \lambda_i\,\mathbf{C}_{00}\,\mathbf{w}_i, \tag{2.48}$$

where $\mathbf{C}_{00}$ is the instantaneous covariance matrix and $\mathbf{C}_{0\tau}$ is the time-lagged cross-covariance matrix. Under the assumptions of the TICA approximation, components associated with the largest eigenvalues approximate the most slowly decorrelating linear collective modes, making them useful candidate sampling coordinates for conformational transitions.

A time-lagged variational autoencoder (TVAE) extends this idea to a non-linear latent space [**Wehmeyer2018**, **Hernndez2018**]. The encoder maps a configuration at time t to a distribution,

$$q_\phi(\mathbf{z}|\mathbf{x}_t) = \mathcal{N}\Big(\boldsymbol{\mu}_\phi(\mathbf{x}_t),\ \sigma^2_\phi(\mathbf{x}_t)\,\mathbf{I}\Big)\,, \tag{2.49}$$

and the decoder reconstructs a later configuration $\mathbf{x}_{t+\tau}$. The time-lagged loss encourages the latent variables to retain information that is predictive over the selected lag time. Deep-TICA similarly replaces the linear TICA map with a neural network,

$$\mathbf{z}_t = f_\theta(\mathbf{x}_t), \tag{2.50}$$

optimized through a time-lagged variational objective so that the learned coordinates approximate slow modes [**Bonati2021**]. These methods are introduced here because Chapter 6 uses PCA, TICA, TVAE, and Deep-TICA as interchangeable learned sampling spaces.

2.5 Supervised Learning for Prediction and Classification

After a representation or latent space is chosen, a model is needed to map it to a property or label. Chapter 3 uses supervised regression for solvation free energy, while Chapter 4 uses classification for ice-phase assignment. The methods below are included because they test whether the chosen representation contains enough information for prediction or state labeling.

2.5.1 Gradient-Boosted Decision Trees: XGBoost

XGBoost is a scalable gradient-boosting framework based on ensembles of decision trees [**Chen2016**]. The prediction for sample i is written as

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \qquad f_k \in \mathcal{F}, \tag{2.51}$$

where $\mathcal{F}$ is the space of regression trees. Trees are added sequentially to reduce residual errors, and XGBoost minimizes a regularized objective,

$$\mathcal{L} = \sum_{i=1}^{M} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k). \tag{2.52}$$

In Chapter 3, XGBoost tests whether non-linear combinations of descriptors and fingerprints can improve solvation prediction while preserving some feature-level interpretability.

2.5.2 Random Forest Regressor

Random Forest Regression (RFR) fits an ensemble of decision tree models based on bootstrap samples and averages their predictions [**Svetnik2003**].

$$\hat{y}(\mathbf{x}) = \frac{1}{T} \sum_{t=1}^{T} h_t(\mathbf{x}), \tag{2.53}$$

The method reduces variance compared to single trees of decisions, and provides a stable baseline for descriptor- and fingerprint-based solvation models. In Chapter 3 it is used to test robust non-linear regression without sequential residual fitting.

2.5.3 Kernel Ridge Regression

Kernel ridge regression (KRR) combines ridge regularization with kernel-based non-linear similarity modeling [**Vovk2013**, **Exterkate2013**]. The primal ridge objective is

$$\min_{\mathbf{w}} \sum_{i=1}^{M} \left(y_i - \mathbf{w}^T \mathbf{x}_i \right)^2 + \lambda \|\mathbf{w}\|^2. \tag{2.54}$$

Using a kernel $k(\mathbf{x}_i, \mathbf{x}_j)$, predictions can be written as

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i k(\mathbf{x}, \mathbf{x}_i), \tag{2.55}$$

with

$$\boldsymbol{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1}\mathbf{y}. \tag{2.56}$$

KRR is useful for moderately sized molecular datasets where chemically similar molecules are expected to have similar properties. Its quadratic memory and cubic direct-solve cost limit scaling to very large datasets. The supervised learning methods described above are compared in Fig. 2.5.

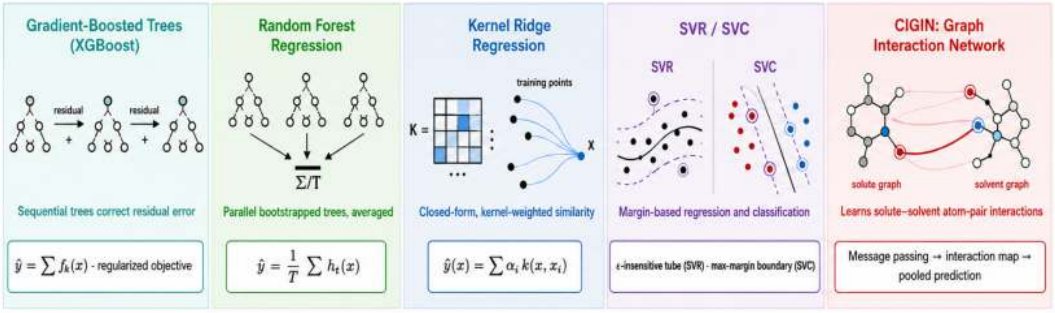

Figure 2.5: Comparison of five supervised machine-learning approaches for molecular property prediction and classification. From left to right, the figure illustrates gradient-boosted trees (XGBoost), random forest regression, kernel ridge regression (KRR), support vector regression/classification (SVR/SVC), and the Chemically Interpretable Graph Interaction Network (CIGIN). Although all methods learn the same mapping from molecular descriptors to target properties, they differ in their underlying learning strategies, ranging from ensemble tree-based methods and kernel-based similarity models to margin-based learning and graph neural networks that explicitly model solute–solvent atom-pair interactions. The schematic highlights the progression from conventional feature-vector methods to graph-native representations capable of learning chemically meaningful intermolecular interactions.

2.5.4 Support Vector Regression and Support Vector Classification

Support vector regression (SVR) fits a function that is as flat as possible while allowing errors inside an ϵ-insensitive tube [**Jingqing2006**]. A common primal form is

$$\min_{\mathbf{w},b,\xi_i,\xi_i^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{M}(\xi_i + \xi_i^*) \tag{2.57}$$

subject to

$$y_i - \mathbf{w}^T\phi(\mathbf{x}_i) - b \le \epsilon + \xi_i, \tag{2.58}$$

$$\mathbf{w}^T\phi(\mathbf{x}_i) + b - y_i \le \epsilon + \xi_i^*, \tag{2.59}$$

$$\xi_i, \xi_i^* \ge 0. \tag{2.60}$$

SVR is used in Chapter 3 as a regularized kernel model for solvation free-energy prediction.

Support vector classification (SVC) uses the same margin-based principle for labels [**Cortes1995**]. In binary classification,

$$f(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{M} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b\right). \tag{2.61}$$

In Chapter 4, SVC is used after SOAP–VAE dimensionality reduction to convert latent-space regions into discrete phase assignments. This separates the representation-learning step from the final phase-labeling step.

2.5.5 Chemically Interpretable Graph Interaction Network

The Chemically Interpretable Graph Interaction Network (CIGIN) represents both solute and solvent as molecular graphs and learns their interaction patterns directly [**Pathak2020**, **Pathak2021**]. In Chapter 3, CIGIN tests whether graph-native solute–solvent interaction learning improves transferability beyond descriptor and fingerprint baselines.

CIGIN first updates atom features through message passing,

$$\mathbf{h}_i^{(t+1)} = U\left(\mathbf{h}_i^{(t)}, \sum_{j \in \mathcal{N}(i)} M\left(\mathbf{h}_i^{(t)}, \mathbf{h}_j^{(t)}, \mathbf{e}_{ij}\right)\right), \tag{2.62}$$

where $\mathbf{h}_i^{(t)}$ is the atom representation at message-passing step t, $\mathbf{e}_{ij}$ is a bond feature vector, and M and U are learned message and update functions. It then constructs pairwise solute–solvent interaction features, schematically written as

$$I_{ij} = \left(\mathbf{h}_i^{\text{solute}}\right)^T \mathbf{h}_j^{\text{solvent}}. \tag{2.63}$$

The resulting interaction map can be inspected to identify atom pairs that influence the prediction. For solvation free energy, that interpretability matters because the target reflects distributed electrostatic, hydrogen-bonding, steric, and dispersion contributions.

2.6 Rare-Event and Path-Based Methods

Rare-event methods are needed when molecular transitions occur on timescales much longer than the MD timestep and are separated by free-energy barriers. Enhanced sampling was introduced conceptually in Section 1.7. This section focuses on the path-based methods used in Chapter 5.

2.6.1 Weighted Ensemble Simulations

The weighted ensemble (WE) method enhances sampling without modifying the Hamiltonian or applying a dynamical bias [**Huber1996**, **Zuckerman2017**, **Aristoff2018**]. It is designed for rare events in which direct MD spends most of its time in stable basins and produces too few complete transitions for reliable kinetic analysis. Instead of changing the potential energy surface, WE changes how computational effort is distributed across an ensemble of unbiased trajectory fragments.

At any WE iteration, the probability distribution is represented by a collection of trajectories, or walkers, each with statistical weight w_i. The weights satisfy

$$\sum_{i=1}^{N_w} w_i = 1, \tag{2.64}$$

where N_w is the number of walkers. For any observable $O(\mathbf{R})$, the ensemble estimate is the weighted average

$$\langle O \rangle \approx \sum_{i=1}^{N_w} w_i O(\mathbf{R}_i), \tag{2.65}$$

where $\mathbf{R}_i$ is the current configuration of walker i. The weights therefore encode probability, while the walker coordinates encode where that probability is located in configuration space.

Each WE iteration has two stages. First, every walker is propagated independently under unbiased dynamics for a fixed iteration time τ_{WE}. Second, the resulting walkers are assigned to bins in a chosen progress-coordinate space and resampled within each bin. The progress coordinate can be a simple geometric coordinate, a pair of CVs, or a PathCV. WE can only allocate walkers efficiently to regions that are resolved by the binning scheme; poor binning wastes unbiased dynamics without formally biasing the result.

If the progress coordinate is denoted $\xi(\mathbf{R})$, then bin membership is defined by

$$b_i = B[\xi(\mathbf{R}_i)], \tag{2.66}$$

where $B[\cdot]$ maps a configuration to a discrete bin. The total probability in bin b is

$$W_b = \sum_{i \in b} w_i. \tag{2.67}$$

Resampling changes the number of walkers in each bin while preserving W_b.

Walker splitting. If a walker of weight w is divided into n children, each child receives

$$w' = \frac{w}{n}. \tag{2.68}$$

The children inherit the parent configuration at the resampling point and subsequently diverge because their stochastic momenta, thermostat variables, or randomized velocity components lead to different unbiased trajectory realizations.

Walker merging. If walkers with weights $\{w_j\}$ are merged, one trajectory survives with probability proportional to its weight and receives

$$w_{\text{new}} = \sum_j w_j. \tag{2.69}$$

For example, walker k is selected with probability

$$p_k = \frac{w_k}{\sum_j w_j}. \tag{2.70}$$

This stochastic selection preserves the expected contribution of the merged walkers to any observable.

The target number of walkers per bin is usually fixed or adaptively controlled. Underpopulated bins are enriched by splitting, while overpopulated bins are compressed by merging. This keeps computational effort distributed across low-probability regions that would otherwise contain few or no trajectories in ordinary MD. Importantly, the resampling step does not alter the microscopic dynamics between resampling events. It only replaces one statistically equivalent weighted representation of the probability distribution with another.

Unbiasedness and Statistical Interpretation

WE is unbiased because propagation and resampling preserve probability in expectation. During propagation, each walker follows the same dynamics that would be used in conventional MD. During resampling, splitting exactly partitions a walker's probability weight, and merging preserves the total weight of the merged set. Consequently, the weighted ensemble remains a valid estimator of the underlying probability distribution.

This distinction matters for the thesis. Biased enhanced sampling methods can accelerate barrier crossing by changing the dynamics and then reweighting. WE instead retains the original dynamics, making it suitable for kinetic observables such as rates, fluxes, and pathway probabilities. The cost is that WE requires a progress coordinate and binning strategy good enough to place walkers near rare transition regions. Poor coordinates do not necessarily bias the result, but they can make the

calculation inefficient. The WE protocol is illustrated schematically in Fig. 2.6.

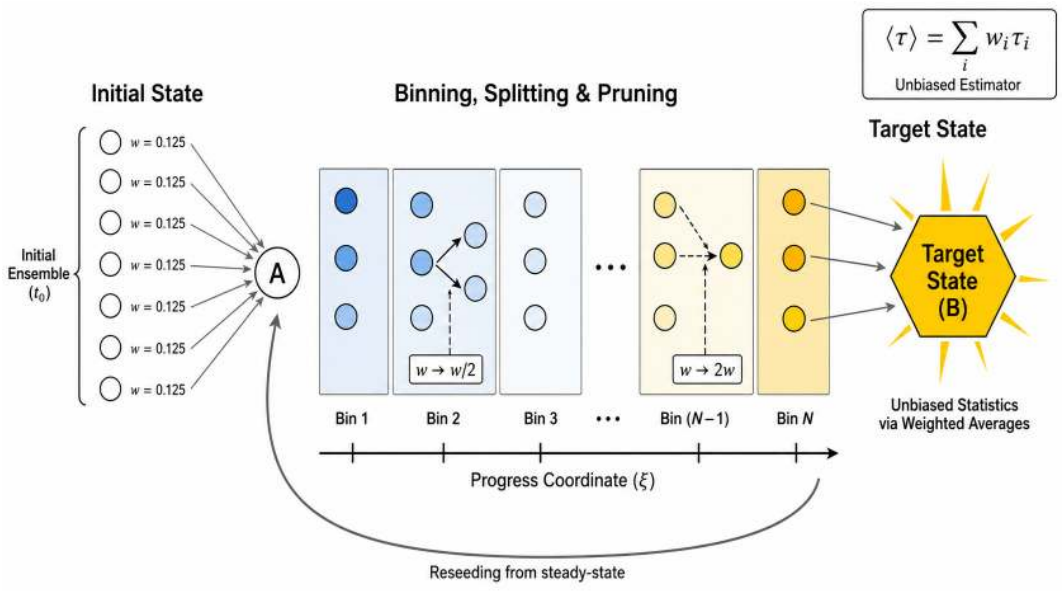

Figure 2.6: Schematic of Weighted Ensemble sampling. Walkers are propagated under unbiased dynamics and periodically resampled within bins along a progress coordinate. Splitting increases resolution in under-sampled regions, merging controls computational cost, and statistical weights preserve the correct ensemble. In steady-state calculations, probability flux into a target state yields kinetic information.

Steady-State Weighted Ensemble Simulations

Steady-state WE estimates rates between predefined source and target states. Consider metastable states A and B. Walkers are initialized in A and propagated using the WE protocol. When a walker reaches B, its weight contributes to the transition flux and the walker is recycled to the source state with the same weight. Recycling maintains a non-equilibrium steady state with a continuous probability current from A to B.

At WE iteration n, the instantaneous probability flux into B can be estimated as

$$J_{AB}^{(n)} = \frac{1}{\tau_{\mathrm{WE}}} \sum_{i \in B \text{ at } n} w_i^{(n)}, \tag{2.71}$$

where the sum includes walkers that enter the target during that iteration. After an initial relaxation period, the running average of this flux approaches a steady value,

$$\bar{J}_{AB} = \frac{1}{N_{\mathrm{iter}} \tau_{\mathrm{WE}}} \sum_{n=1}^{N_{\mathrm{iter}}} \sum_{i \in B \text{ at } n} w_i^{(n)}. \tag{2.72}$$

Under steady-state conditions, the rate constant follows from the Hill relation,

$$k_{AB} = \frac{J_{AB}}{P_A}, \tag{2.73}$$

where J_{AB} is the probability flux from A to B and P_A is the equilibrium population of the source basin. In practice, the flux is estimated from cumulative recycled weight,

$$J_{AB} = \frac{1}{t_{\mathrm{sim}}} \sum_{\mathrm{recycle}} w_i, \tag{2.74}$$

where t_{sim} is the accumulated simulation time. If the source population is normalized to unity by recycling, P_A is effectively maintained by construction and the steady flux directly determines the rate.

WE also supports pathway-resolved kinetic analysis. If target-reaching walkers are assigned to pathway class α, the pathway-specific flux can be written as

$$J_{AB}^{(\alpha)} = \frac{1}{t_{\mathrm{sim}}} \sum_{\mathrm{recycle}\in\alpha} w_i. \tag{2.75}$$

The relative contribution of pathway α is then

$$p_\alpha = \frac{J_{AB}^{(\alpha)}}{\sum_\beta J_{AB}^{(\beta)}}. \tag{2.76}$$

This is the reason WE is paired with path identification in Chapter 5: the same weighted trajectory ensemble can provide both rate estimates and mechanistic decomposition into competing pathways [**Hill1989**, **Suarez2014**, **Bhatt2010**].

Practical Requirements and Failure Modes

The main practical inputs to a WE calculation are the initial ensemble, the source and target definitions, the progress coordinates, the binning scheme, the target number of walkers per bin, and the iteration time τ_{WE}. These choices do not change the formal dynamics, but they determine whether the simulation spends effort in useful regions of configuration space.

Three failure modes are especially relevant here. First, bins that do not resolve the transition mechanism may concentrate walkers in regions that are geometrically distinct but kinetically unimportant. Second, overly fine bins can fragment probability so strongly that many bins contain poorly sampled walkers. Third, pathway heterogeneity can mix mechanistically distinct routes unless the progress coordinate or post-analysis separates them. These are the practical reasons Chapter 5 combines path identification

with WE rather than treating binning as a purely geometric choice.

2.6.2 Path Collective Variables

Path collective variables (PathCVs) describe progress along a curved transition pathway rather than distance from a single reference structure [**Branduardi2007**]. A PathCV is defined using a sequence of reference conformations

$$\{X_i\}_{i=1}^{N_p}, \tag{2.77}$$

connecting reactant and product states. For a configuration X, the distance to reference i is

$$D_i = R(X, X_i), \tag{2.78}$$

where R is a structural or latent-space distance. The progress coordinate is

$$s(X) = \frac{\sum_{i=1}^{N_p} i \exp(-\lambda D_i)}{\sum_{i=1}^{N_p} \exp(-\lambda D_i)}, \tag{2.79}$$

and the orthogonal displacement is

$$z(X) = -\frac{1}{\lambda} \ln \left[\sum_{i=1}^{N_p} \exp(-\lambda D_i) \right]. \tag{2.80}$$

The coordinate $s(X)$ measures progress along the pathway, while $z(X)$ measures deviation away from it. In Chapter 5, transition-path ensembles are clustered and refined into PathCVs, which then define pathway-specific WE simulations.

2.6.3 Dynamic Time Warping for Transition Path Analysis

Dynamic time warping (DTW) compares transition trajectories whose structural progress occurs at different rates [**Sakoe1978**]. Given two trajectories

$$A = \{a_1, \ldots, a_m\}, \tag{2.81}$$

and

$$B = \{b_1, \ldots, b_n\}, \tag{2.82}$$

DTW identifies an alignment path π that minimizes

$$D_{\mathrm{DTW}}(A, B) = \min_{\pi} \sum_{(p,q)\in\pi} d(a_p, b_q), \tag{2.83}$$

where $d(a_p, b_q)$ is the distance between points in CV or structural space. DTW allows local stretching and compression of the time axis, so trajectories following the same pathway can be grouped even if they dwell in intermediates for different durations. Chapter 5 uses DTW distances to cluster transition trajectories into mechanistic pathway families.

2.7 Markov State Modeling and Kinetic Analysis

Markov state models (MSMs) convert ensembles of molecular trajectories into long-timescale kinetic models [**Prinz2011**, **Chodera2014**, **MSM2014**]. They are especially useful when the simulation data consist of many shorter trajectories rather than one fully converged trajectory. In that setting, adaptive sampling improves configurational coverage, while the MSM stage estimates transition probabilities, stationary populations, and kinetic observables from longer unbiased trajectory data.

An MSM addresses the sampling bottleneck from a different direction than WE. WE estimates rare-event fluxes by propagating weighted trajectories, whereas an MSM estimates the transition operator from trajectory data after the fact. The two approaches therefore have different requirements. WE requires useful progress bins during sampling; an MSM requires enough transition counts between discretized states at a lag time for which the dynamics are approximately Markovian.

2.7.1 State Discretization

The first step in MSM construction is to choose a molecular representation and discretize it into states. A trajectory frame is mapped from coordinates $\mathbf{R}_t$ to a feature vector $\mathbf{x}_t$, then optionally to a lower-dimensional coordinate $\mathbf{z}_t$ using PCA, TICA, VAE, TVAE, or another projection:

$$\mathbf{R}_t \rightarrow \mathbf{x}_t \rightarrow \mathbf{z}_t. \tag{2.84}$$

The projected data are then partitioned into microstates using a clustering or gridding procedure. If $s_t \in \{1, \ldots, n\}$ denotes the discrete state assigned to frame t, the continuous trajectory becomes a state trajectory,

$$\{\mathbf{R}_t\}_{t=1}^{T} \rightarrow \{s_t\}_{t=1}^{T}. \tag{2.85}$$

This discretization is the main point at which representation quality enters MSM analysis. If the features fail to resolve the slow processes, or if the state decomposition merges kinetically distinct basins, the resulting model may appear statistically well behaved while still missing the relevant mechanism.

2.7.2 Transition Count and Transition Probability Matrices

For a chosen lag time τ, transition counts are accumulated as

$$C_{ij}(\tau) = \sum_t \mathbf{1}\left[s_t = i,\ s_{t+\tau} = j\right], \tag{2.86}$$

where $\mathbf{1}[\cdot]$ is an indicator function. A simple maximum-likelihood transition matrix is obtained by row normalization,

$$T_{ij}(\tau) = \frac{C_{ij}(\tau)}{\sum_k C_{ik}(\tau)}. \tag{2.87}$$

Each element $T_{ij}(\tau)$ estimates the probability of being in state j after time τ given that the system started in state i. For equilibrium MD, reversible estimators are often used so that the transition matrix satisfies detailed balance,

$$\pi_i T_{ij} = \pi_j T_{ji}, \tag{2.88}$$

where $\boldsymbol{\pi}$ is the stationary distribution. The stationary distribution is the normalized left eigenvector satisfying

$$\boldsymbol{\pi}^T \mathbf{T} = \boldsymbol{\pi}^T. \tag{2.89}$$

Once $\boldsymbol{\pi}$ is known, free-energy-like profiles in a projected coordinate can be estimated from

$$F_i = -k_B T \ln \pi_i + \text{constant}. \tag{2.90}$$

This type of MSM reweighting is used in Chapter 6 to interpret conformational populations from production trajectories initialized across an explored landscape. The MSM construction workflow is illustrated in Fig. 2.7.

2.7.3 Markovianity and Implied Timescales

The central approximation in an MSM is that, at lag time τ, the future state depends only on the present state and not on the earlier history:

$$P(s_{t+\tau} = j \mid s_t = i, s_{t-\tau}, \ldots) \approx P(s_{t+\tau} = j \mid s_t = i). \tag{2.91}$$

This approximation becomes more accurate when τ is long enough for fast intrastate relaxation to decay, but choosing τ too large reduces the number of observed transitions and increases statistical uncertainty.

The eigenvalues of the transition matrix provide a practical diagnostic. If $\lambda_k(\tau)$ is

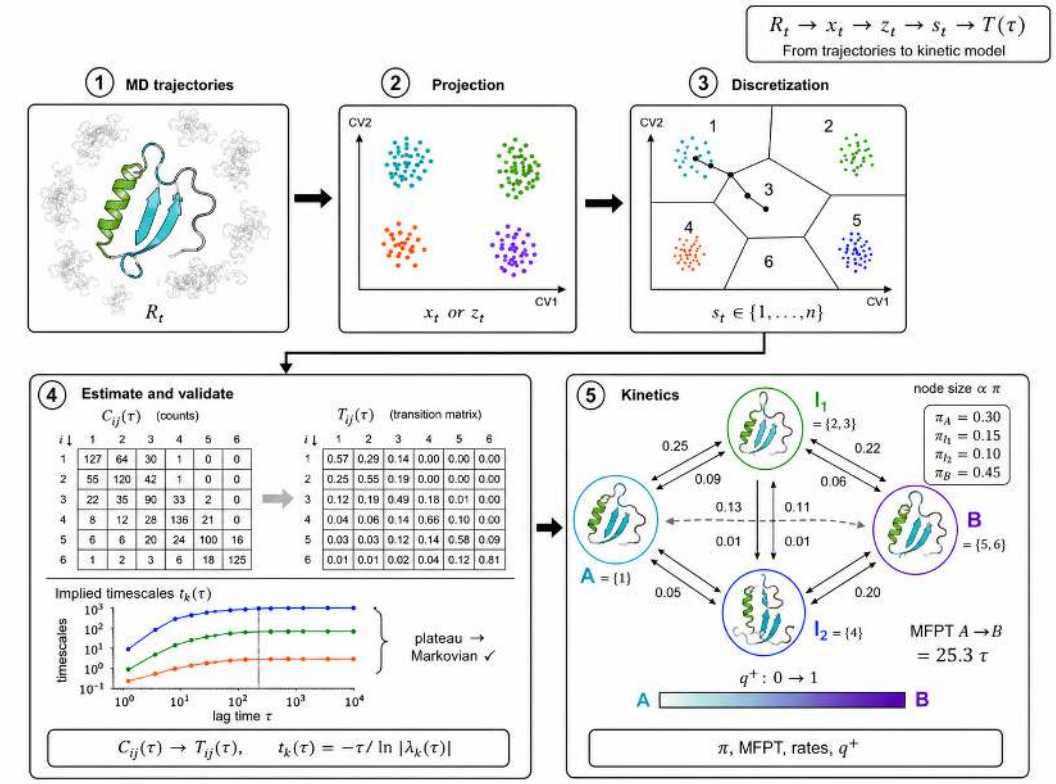

Figure 2.7: Workflow for constructing and analyzing a Markov state model (MSM) from molecular dynamics trajectories. Raw trajectory coordinates (R_t) are projected into a lower-dimensional feature space ((x_t) or (z_t)) and discretized into microstates (s_t). Transition counts ($C_{ij}(\tau)$) are accumulated at lag time (τ) and row-normalized to obtain the transition probability matrix ($T_{ij}(\tau)$). Markovianity is assessed through implied-timescale convergence, after which the kinetic model is coarse-grained into macrostates and represented as a state network. The resulting MSM provides stationary populations (π), transition rates, mean first-passage times (MFPTs), and committor probabilities (q^+) for quantitative analysis of metastable-state interconversion.

the kth eigenvalue of $\mathbf{T}(\tau)$, the corresponding implied relaxation timescale is

$$t_k(\tau) = -\frac{\tau}{\ln|\lambda_k(\tau)|}. \tag{2.92}$$

For a well-resolved Markovian model, the dominant implied timescales should become approximately independent of τ over a suitable lag-time range. Chapter 6 uses this criterion to select the lag time for kinetic analysis. Failure to reach a plateau usually indicates inadequate state discretization, insufficient sampling, or hidden slow variables not captured by the chosen representation.

2.7.4 Chapman–Kolmogorov Validation

A second validation test is the Chapman–Kolmogorov relation. If the model is Markovian at lag time τ, then transition probabilities over a longer time $m\tau$ should be approximated by repeated matrix multiplication:

$$\mathbf{T}(m\tau) \approx \mathbf{T}(\tau)^m. \tag{2.93}$$

In practice, one compares transition probabilities predicted from $\mathbf{T}(\tau)^m$ with those estimated directly from trajectory data at lag $m\tau$. Agreement supports the chosen discretization and lag time; disagreement indicates memory effects or insufficient sampling. This test is particularly important when the input trajectories are seeded from non-equilibrium or adaptively selected structures, because the production trajectories must relax sufficiently before being interpreted kinetically.

2.7.5 Kinetic Observables

Once a validated transition matrix is available, long-timescale observables can be computed without requiring every transition to be observed repeatedly in a single trajectory. For metastable sets A and B, the mean first passage time (MFPT) from A to B can be obtained by solving a system of linear equations. For each state $i \notin B$,

$$m_i = \tau + \sum_{j \notin B} T_{ij} m_j, \tag{2.94}$$

with boundary condition $m_i = 0$ for $i \in B$. The MFPT from an initial distribution over A is then the corresponding population-weighted average of m_i over $i \in A$.

The same transition matrix can be used to estimate committor probabilities, dominant transition pathways, and fluxes between metastable sets. The forward committor q_i^+,

defined as the probability of reaching B before returning to A from state i, satisfies

$$q_i^+ = \sum_j T_{ij} q_j^+, \tag{2.95}$$

with boundary conditions $q_i^+ = 0$ for $i \in A$ and $q_i^+ = 1$ for $i \in B$. These quantities connect MSM analysis to the pathway and rare-event questions addressed elsewhere in the thesis.

2.7.6 Coarse-Graining and Interpretation

Microstate MSMs are often too detailed to interpret directly. Coarse-graining groups microstates into macrostates that correspond to metastable basins such as folded and unfolded conformations, bound and unbound states, or major structural intermediates. The coarse-grained model should retain the slow transition structure while reducing the state space enough to report populations, transition probabilities, MFPTs, and mechanism-level summaries.

The quality of a coarse-grained MSM depends on the same choices that affect the microstate model: representation, state decomposition, lag time, and sampling coverage. A two-state model may be appropriate when the dominant process is a simple folded–unfolded transition, but it can hide intermediates or parallel pathways if the underlying dynamics are more complex. For this reason, MSMs should be treated as kinetic models whose validity must be checked, not just as visualization tools.

2.7.7 Relation to Variational Kinetic Models

MSMs are closely related to variational approaches for learning slow dynamics. TICA can be interpreted as a linear approximation to the leading eigenfunctions of the transfer operator, and VAMP-based methods generalize this idea to non-reversible dynamics and neural-network representations [**No2013**, **Wu2020VAMP**]. This connection explains why time-lagged projections are useful before MSM construction: they attempt to preserve the slow coordinates that dominate long-time kinetics.

In this dissertation, MSMs are used after representation learning, state identification, or exploratory sampling has produced candidate regions of configuration space. The MSM step converts adequately sampled trajectory data into populations, relaxation timescales, and transition rates. This separation matters because exploratory trajectories can suggest where important conformations are located, but quantitative MSM estimates should be based on trajectory data that satisfy the assumptions of the Markov model.

3

Transferability of Molecular Representations for Aqueous Solvation Free-Energy Prediction

A model that predicts solvation free energy accurately on a random test split is not necessarily useful beyond the dataset on which it was trained. This chapter therefore asks a stricter question: *which molecular representations preserve predictive accuracy when the chemical distribution changes?* We compare physicochemical descriptors, fingerprints, molecular graphs, and a trajectory-derived molecular representation for predicting aqueous solvation free energy (ΔG_{solv}). The analysis proceeds from simple and interpretable representations to increasingly expressive learned models, while tracking the physical failure modes that remain at each stage. Particular attention is given to whether conformational information and explicit applicability-domain descriptors improve prediction for molecules outside the training chemistry.

3.1 Motivation and Physical Context

The solvation free energy is the thermodynamic work needed to transfer a solute molecule from the gas phase into a solvent at constant temperature and pressure. The experimental values used here are treated as curated aqueous transfer free energies under the conventions of their source datasets; no attempt is made to recompute all measurements under a single standard-state convention. The transfer can be expressed within the Ben-Naim coupling framework, where the interactions of the solute with the solvent are turned on continuously using a coupling parameter $\lambda \in [0, 1]$ [**Mobley2014**]. The process can be separated into three physical contributions:

1. **Cavity formation:** the work needed to displace solvent molecules and form a cavity of suitable size and shape in the bulk liquid.
2. **Van der Waals interactions:** dispersion and short-range repulsion between the solute and the solvent within the cavity.
3. **Electrostatic charging:** polarization of the solvent by the solute charge distribution.

Evaluating these stages directly requires high-level quantum chemistry or molecular dynamics simulations. Density functional theory, quantum chemical solvation models

such as PCM or SMx, and free-energy perturbation simulations provide physically grounded estimates of ΔG_{solv}, but their computational cost limits their use in high-throughput screening [**Kang2012**, **Cavalli2006**, **Spiegel2006**, **Senn2007**, **Senn2009**, **Cardoso2007**, **Barua2019**]. This bottleneck has driven the development of QSPR models that predict solvation free energy directly from molecular structure.

Aqueous solvation free energy is also closely connected to molecular recognition, partitioning, solubility, and transport. These links make ΔG_{solv} a useful test case for molecular machine learning: it is experimentally accessible, physically interpretable, and sensitive to molecular size, polarity, hydrogen bonding, and conformational exposure. A successful model must therefore learn more than a statistical association with a few familiar functional groups; it must retain the relevant physical trends when evaluated on new chemical space [**KALYAANAMOORTHY2011831**, **Meng2011**, **Pantaleao**, **Sanachai2022**, **Lin2018**, **Jasuja2014**, **Mafethe2023**, **Giordano2022**, **Voet2013**, **Seidel2020**, **Kaserer2015**, **Muhammed2018**, **Reichel2015**, **Jang2001**, **Sharma2021**, **Lai2022**].

Modern machine learning has expanded QSPR modeling by learning nonlinear mappings from molecular features to properties [**Mitchell2014**, **Niazi2023**, **Lo2018**, **JimnezLuna2021**, **Priya2022**, **Paul2021**, **GoelEfficient2023**]. Public databases provide the experimental values needed to train these models [**Rutz2021**, **Sorokina2020**, **Banerjee2014**, **Zeng2017**, **Xue2012**, **Mendez2018**, **Gilson2015**, **Wishart2017**, **Siramshetty2021**, **Sussman1998**]. However, the accuracy and transferability of such models depend strongly on the representation used to encode each molecule.

We compare four representation families:

1. **Physicochemical descriptors:** low-dimensional vectors of calculated properties such as polar surface area, hydrogen-bond counts, and logP.

2. **Molecular fingerprints:** fixed-length binary vectors encoding the presence or absence of structural subgraphs.

3. **Graph encodings:** non-Euclidean representations that preserve molecular topology, with atoms as nodes and bonds as edges.

4. **Trajectory-derived learned embeddings:** continuous molecular vectors learned from molecular dynamics trajectories, optionally augmented with physically interpretable descriptors and applicability-domain flags.

Each representation has different information content, interpretability, and extrapolation behavior [**Haghighatlari2020**, **David2020**]. Random train-test splits can mask weaknesses in generalization when training and test molecules are chemically similar. A stricter test is transferability: a model trained on one dataset should retain accuracy

when evaluated on a chemically distinct external dataset. It is not enough to validate only internally, we need to validate externally on different datasets to ensure the model has learned relationships that are transferable in the real world and not just correlations specific to the dataset [**Llompart2024**].

3.2 Datasets and Curation with Leakage Awareness

Three datasets of aqueous solvation free energies were collected: Minnesota Solvation Database (MNSol), FreeSolv and CombiSolv-Exp-8780 [**mnsol**, **Mobley2014**, **Vermeire2021**, **YuSolvbert2023**]. The datasets vary in scale, chemical diversity, and measurement protocol, making them useful for probing model generalization. The reconciled water-only inventory used for the transfer study contained 642 FreeSolv entries, 1029 CombiSolv entries, and 312 MNSol entries. After neutral-molecule filtering and out-of-domain filtering, the effective benchmark sizes were 642 FreeSolv molecules, 599 CombiSolv molecules, and 65 MNSol molecules.

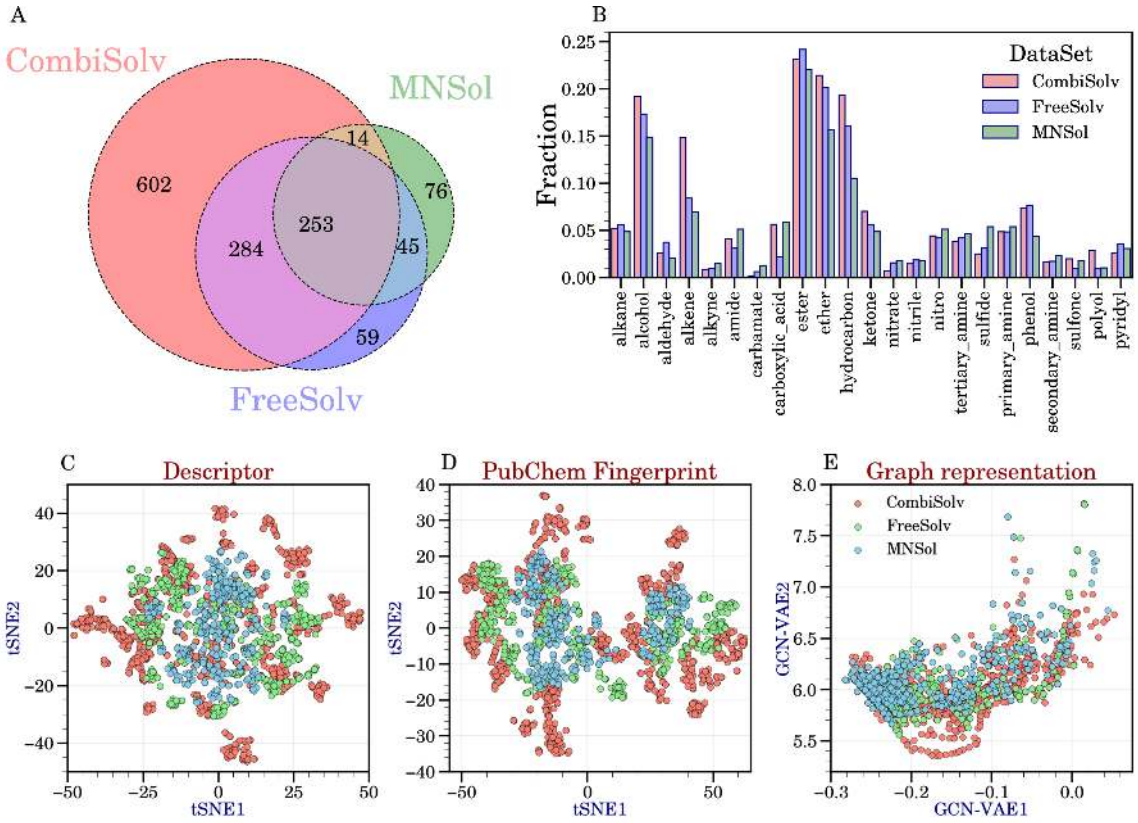

Figure 3.1: Overview of the solvation free-energy datasets used in this chapter. (A) Overlap of chemical entities among MNSol, FreeSolv, and CombiSolv. (B) Functional-group populations in each dataset. (C,D) Two-dimensional t-SNE projections of descriptor and PubChem fingerprint representations. (E) Two-dimensional latent embedding learned from molecular graphs using a GCN-VAE model.

The trajectory-derived transfer study used FreeSolv for training and evaluated

transfer on filtered subsets of CombiSolv and MNSol. To prevent chemical overlap from inflating performance, molecules sharing the first InChIKey block with a FreeSolv entry were removed. We also excluded near-neighbour analogues whose Morgan-fingerprint Tanimoto similarity to any FreeSolv training molecule exceeded 0.85. CombiSolv was then separated into a diagnostic development set and a held-out test set, with the same near-neighbour filtering applied across the split. The final split contained 345 CombiSolv development molecules and 254 held-out CombiSolv test molecules. The development split retained 82 connectivity overlaps and 142 high-similarity analogues for model-development diagnostics, whereas the held-out test split contained no such overlaps by the same criteria. MNSol was retained as a 65-molecule external stress test; because inspection of the MNSol errors informed the later design of outlier-aware descriptors, it is interpreted as a diagnostic stress test rather than as fully blind confirmatory evidence.

3.3 A Progression of Molecular Representations

Machine-learning models cannot operate directly on a SMILES string; each molecule must first be mapped to a numerical representation. We examine four levels of representation: global physicochemical descriptors, fragment-based fingerprints, atomistic molecular graphs, and a trajectory-derived embedding. This ordering is deliberate. Each step adds information that the preceding representation lacks, allowing the transfer failures to be traced from global composition, to local structure, to atomistic topology, and finally to conformational and trajectory-level physics. The trajectory-derived representation follows the general strategy of self-supervised representation learning used by continuous molecular descriptors such as CDDD [**Winter2019**], but replaces two-dimensional translation tasks with supervision derived from three-dimensional molecular trajectories and physical observables.

3.3.1 Descriptor Representation

Each molecule is converted into a vector of computed chemical and physical properties. In this work, 217 descriptors were calculated using RDKit [**rdkit**]. The descriptor set includes topological polar surface area (TPSA), molecular weight, hydrogen-bond donors and acceptors, quantitative estimate of drug-likeness (QED), and octanol-water partition coefficient (logP). To reduce noise and redundancy, descriptors with zero variance were removed, and Pearson correlation filtering was applied so that for descriptor pairs with $|r| > 0.75$, one descriptor was discarded. This preprocessing reduced the descriptor space to 116 dimensions.

3.3.2 Fingerprint Representation

Fingerprints encode structural motifs into fixed-length binary vectors. We evaluate two classes of fingerprints:

1. **Structural keys:** binary vectors where each bit corresponds to a predefined fragment. We use MACCS keys (166 bits) and PubChem fingerprints (881 bits) [**DurantReoptimization2002**, **FernndezdeGortari2017**].

2. **Hashed fingerprints:** dynamically generated fragments mapped to a bit vector using a hash function. We evaluate Atom-pair, RDKit, and Morgan fingerprints [**CarhartAtom1985**, **Morgan1965**, **Rogers2010**].

To balance resolution and sparsity, all hashed fingerprints used a length of 2048 bits.

3.3.3 Graph Representation

Molecules are represented as undirected graphs $G(V, E)$, where atoms are nodes and bonds are edges. Each atom $i \in V$ is assigned a feature vector h_i^0 encoding atomic number, hybridization, formal charge, aromaticity, and valence. Each bond $(i, j) \in E$ is assigned a feature vector encoding bond order, ring membership, and stereochemistry. The connectivity is described by the adjacency matrix

$$A_{ij} = \begin{cases} 1, & \text{if atoms } i \text{ and } j \text{ are bonded,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

The molecular graphs were formatted for the Chemically Interpretable Graph Interaction Network (CIGIN) [**Pathak2020**, **Pathak2021**], which models solute-solvent interactions using atom-level graph features.

3.3.4 Trajectory-Derived Molecular Representation

In addition to static descriptors, fingerprints, and graph encodings, we evaluated a molecular representation learned from molecular dynamics trajectories. The representation-learning model was pretrained on a database of 9371 valid trajectories, each containing multiple molecular conformations, atomic coordinates, covalent connectivity, and frame-level physical observables. The aim of pretraining was not to predict solvation free energy directly, but to construct a coordinate-aware representation that captures conformational geometry, energetic patterns, and global molecular shape. The complete representation-learning and downstream prediction workflow is summarized in Fig. 3.2.

Trajectories were generated directly from SMILES representations using a standardized and restartable simulation pipeline. For each molecule, the input SMILES string

was canonicalized and converted into a three-dimensional conformer using RDKit. Parameterized systems were simulated with OpenMM, and production trajectories, thermodynamic logs, energy-decomposition data, and extracted feature vectors were organized in an InChIKey-indexed filesystem library. This indexing scheme allowed previously completed simulations to be reused across datasets, prevented redundant trajectory generation for duplicate molecules, and enabled interrupted calculations to be resumed without repeating completed stages.

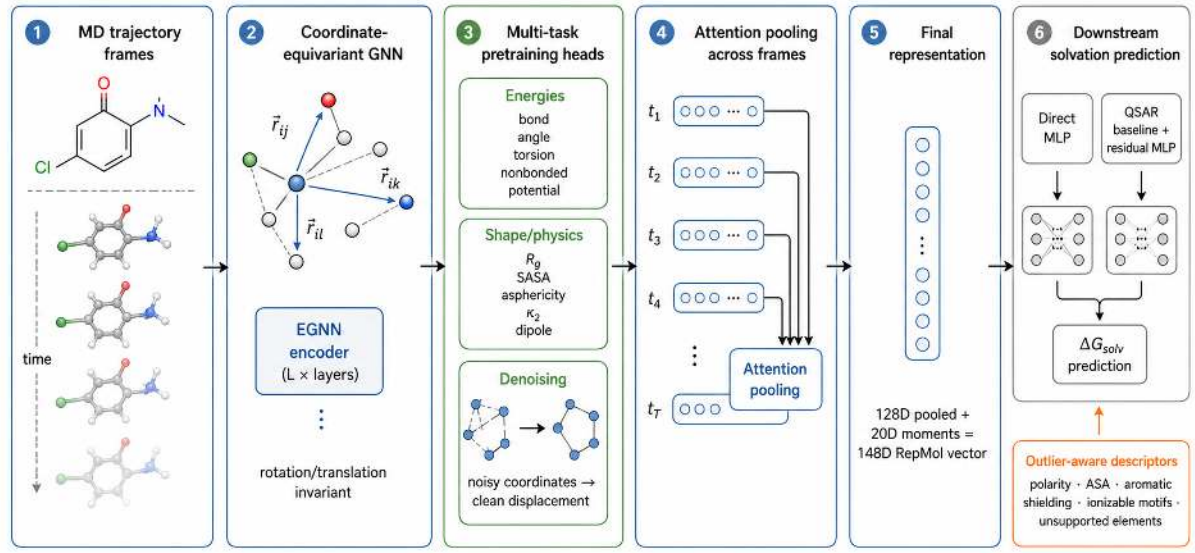

Figure 3.2: Trajectory-based molecular representation-learning workflow. Molecular dynamics frames are encoded using a coordinate-equivariant graph neural network pretrained against energetic components, global shape and physical observables, and a coordinate-denoising objective. Frame-level embeddings are combined through attention pooling, while predicted trajectory observables are summarized by their means and standard deviations. The final 148-dimensional representation combines a 128-dimensional pooled embedding with 20 trajectory-moment features. It is subsequently used either for direct solvation free-energy prediction or to correct a descriptor-based QSAR baseline. Optional outlier-aware descriptors provide information about solvent-accessible polarity, aromatic shielding, potentially ionizable motifs, and unsupported elements.

The trajectory encoder is a four-layer coordinate-equivariant graph neural network. Atoms are represented as nodes, covalent bonds define the graph edges, and interatomic distances are expanded using 20 radial basis functions with a 1.0 nm cutoff. At layer ℓ, messages are computed from neighboring atom features, squared distances, and radial edge features:

$$m_{ij}^{(\ell)} = \phi_m \left(h_i^{(\ell)}, h_j^{(\ell)}, \left| x_i^{(\ell)} - x_j^{(\ell)} \right|^2, e_{ij} \right). \tag{3.2}$$

Node features are updated by aggregating neighboring messages:

$$h_i^{(\ell+1)} = \phi_h\left(h_i^{(\ell)}, \sum_{j\in\mathcal{N}(i)} m_{ij}^{(\ell)}\right). \tag{3.3}$$

The coordinate update is equivariant at the coordinate level, while the pooled molecular representation is designed to be invariant to rigid translations and rotations of the input conformation. This invariance was checked by comparing embeddings computed before and after applying random rigid-body transformations to the input conformations.

The model was pretrained using eleven self-supervised or physics-supervised targets. Five targets corresponded to energetic components: bond, angle, torsion, nonbonded, and total potential energy. Five additional targets described global shape or physical observables: radius of gyration, solvent-accessible surface area, gyration anisotropy, asphericity, and dipole magnitude. The final target was a coordinate-denoising task in which Gaussian noise was added to the atomic coordinates and the model predicted the displacement required to recover the unperturbed structure. The task-specific losses were combined using learnable log-variance weights, allowing the network to balance objectives with different numerical scales and uncertainties.

After pretraining, the EGNN was applied to all frames of each molecular trajectory. A 128-dimensional frame embedding was produced for each conformation and subsequently pooled across frames using attention-based trajectory pooling. In parallel, the predicted physical observables were summarized over the trajectory using their means and standard deviations. Ten predicted observables therefore contributed 20 trajectory-moment features. The final trajectory-derived feature vector was defined as

$$\mathbf{z}_{\text{traj}} = \left[\mathbf{z}_{\text{pool}}^{128}, \mathbf{z}_{\text{mom}}^{20}\right] \in \mathbb{R}^{148}. \tag{3.4}$$

The resulting vector is referred to hereafter as the *trajectory-derived representation*. The predicted trajectory moments are treated as learned physical proxies rather than exact thermodynamic quantities. In particular, their standard deviations quantify variations in the predicted observables across sampled conformations, but they should not be interpreted as direct estimates of conformational entropy.

3.4 Chemical-Space Structure and Representation Maps

To visualize chemical space, we performed principal component analysis (PCA), t-distributed stochastic neighbor embedding (t-SNE), and uniform manifold approximation and projection (UMAP) on descriptor and fingerprint representations [**Pearson1901**, **tsne1**, **tsne2**, **tsne3**, **Healy2024**]. The t-SNE projection of PubChem fingerprints gave the clearest separation of functional classes. Because molecular graphs are non-Euclidean,

we also trained a relational graph convolutional variational autoencoder (GCN-VAE) to project graphs into a continuous latent space. The encoder uses relational graph convolutional layers [**rgcn**, **wang2019dgl**] followed by global pooling, and the decoder reconstructs the molecular graph using a reconstruction loss and KL regularizer [**Belov2011**]. The GCN-VAE embedding in Fig. 3.1E illustrates the chemical diversity of CombiSolv relative to MNSol and FreeSolv.

To inspect whether the trajectory-derived representation organizes chemically meaningful structure, the 148-dimensional vectors were standardized and projected to two dimensions using t-SNE. Molecules were colored by a SMARTS-based functional-group hierarchy including carboxylic acids, esters, amides, alcohols, amines, carbonyls, ethers, halides, aromatics, and aliphatic or alkane molecules. This map is interpreted as a local-neighborhood visualization rather than a quantitative clustering model: nearby points indicate similarity in the learned representation, but cluster boundaries are not used as labels during training.

Figure 3.3 shows that the trajectory-derived vectors preserve local chemical neighborhoods in the visualization, while remaining distinct from the supervised regression models used below.

3.5 Prediction Models and Validation Protocol

The benchmark combines conventional regressors with graph-based and representation-learning models. The purpose is not to rank algorithms in isolation, but to determine how each representation behaves under internal validation and external chemical shift:

1. **XGBoost:** gradient-boosted decision trees on tabular descriptors, with feature-importance analysis [**Chen2016**, **Ghanavati2024**, **Mohammadi2022**, **Yang2024**].
2. **Random Forest Regressor (RFR):** an ensemble of decision trees trained independently [**Svetnik2003**]
3. **Kernel Ridge Regression (KRR) and Support Vector Regression (SVR):** kernel based nonlinear models which are well suited for moderate sized datasets [**Vovk2013**, **Exterkate2013**, **Jingqing2006**].
4. **Multilayer perceptron regressor (MLPR):** a feedforward neural network with three hidden layers trained with Adam [**adam**].
5. **CIGIN GNN:** a graph neural network that combines solute and solvent graph features through an interaction matrix [**Pathak2020**, **Pathak2021**].
6. **Trajectory-informed downstream models:** neural predictors that either estimate ΔG_{solv} directly from the trajectory-derived representation or learn a correction to a descriptor-based QSAR baseline, with optional applicability-domain descriptors.

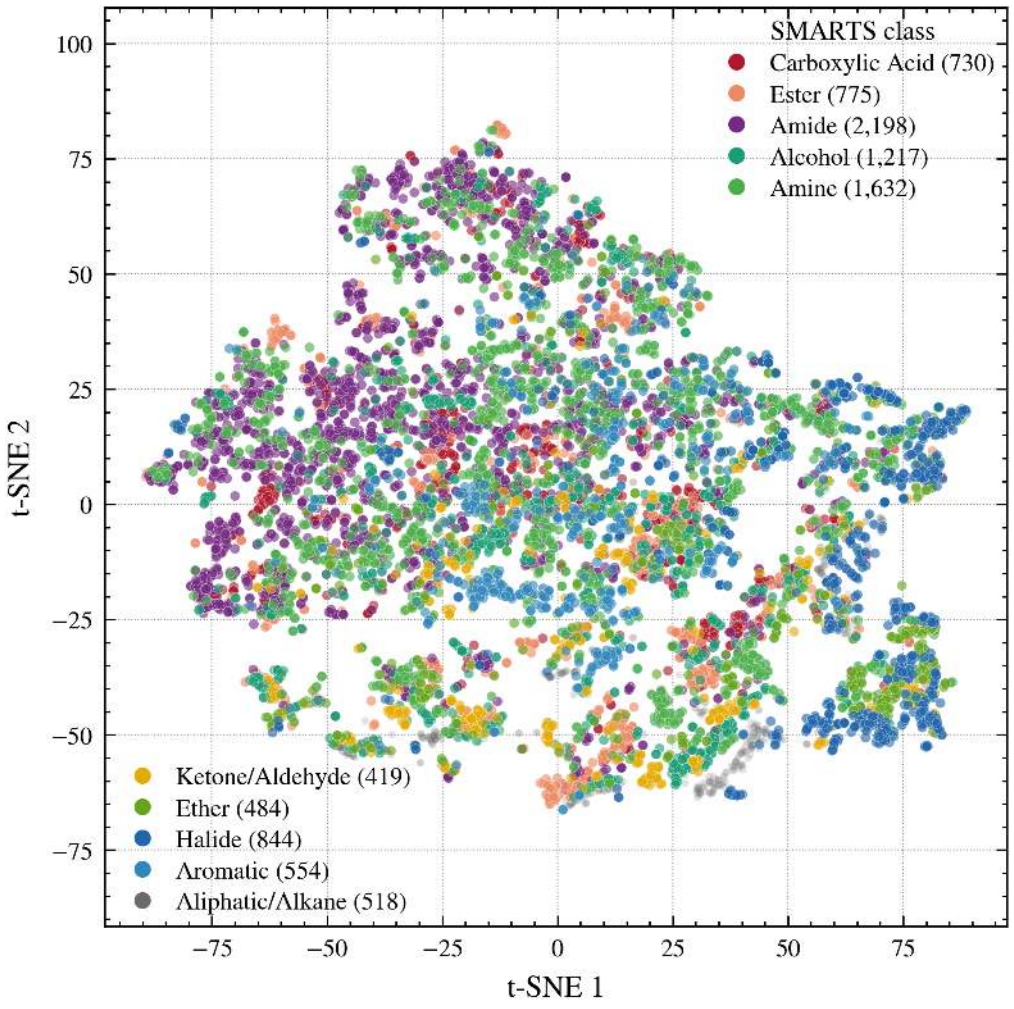

Figure 3.3: t-SNE projection of the 148-dimensional trajectory-derived representation. Points are colored by SMARTS-based functional-group labels for the major chemical classes. The projection shows that the trajectory-derived representation preserves chemically meaningful local structure, but the two-dimensional map is used only for visualization and not as a supervised model input or a quantitative clustering assignment.

Two downstream formulations were used to test the value of the trajectory-derived representation. In the *direct trajectory-informed model*, a multilayer perceptron predicts ΔG_{solv} directly from the learned representation. In the *trajectory-informed residual model*, a Ridge-regression QSAR estimate is first obtained from two-dimensional descriptors, after which a neural network learns the remaining prediction error using the trajectory-derived representation, the baseline estimate, and the descriptor vector.

A second set of models additionally receives outlier-aware descriptors that encode solvent-accessible polarity, aromatic planarity, fused-ring structure, heteroatom density, potentially ionizable neutral motifs, unsupported elements, and rare chemotypes such as anthraquinones, quinones, sulfonylureas, sulfonamides, carboxylic acids, and organosilicon compounds. These descriptors do not impose manual corrections; they provide information about situations in which conventional structural descriptors may be incomplete or misleading.

Model performance is assessed using the coefficient of determination R^2, mean squared error (MSE), and mean absolute error (MAE):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{3.5}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{3.6}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{3.7}$$

Here y_i is the experimental value, $\hat{y}_i$ is the prediction, and $\bar{y}$ is the experimental mean. Regressors were implemented in Scikit-Learn [**scikit_learn**]. We performed both internal k-fold cross-validation and external cross-dataset transfer tests.

3.6 Results: What Each Representation Learns and Misses

The results are organized as a progression rather than as a collection of unrelated benchmarks. Global descriptors provide an interpretable starting point; fingerprints add local substructure; graph models add atom-level connectivity; and trajectory-derived features add conformational and trajectory-level information. At each stage, the central question is the same: which transfer failures are resolved, and which remain? The recurring tests involve molecular-size extrapolation, cavity formation, inaccessible or shielded polarity, rare linkers, and chemotypes that are weakly represented or unsupported in the training data.

Table 3.1: Progressive gains and remaining transfer limitations across the four representation families. Each representation resolves part of the preceding limitation, but no representation eliminates all forms of chemical-distribution shift. MAE values are reported in kcal mol^{-1}, whereas MSE values are reported in $(\text{kcal mol}^{-1})^2$.

Representation	**What it adds**	**Observed transfer behavior**	**Main limitation**
Global descriptors	Molecular size, polarity, and hydrogen-bond counts	Strong internal trends, but sensitivity to changes in chemical range and composition	Limited information about the local environment and accessibility of polar groups
Fingerprints	Local fragments and substructure patterns	Better discrimination of chemical motifs; a neural model reduced the Morgan-fingerprint MSE on CombiSolv from 2.2 to 1.1 $(\text{kcal mol}^{-1})^2$	Fragment occurrence does not encode the physical scaling of cavity formation or solvent response
Molecular graphs	Atom-level connectivity and solute–solvent interactions	Strong within-dataset performance, but MNSol-to-CombiSolv transfer reached an MSE of 5.48 $(\text{kcal mol}^{-1})^2$	Poor extrapolation for long alkanes, macrocycles, and other underrepresented structures
Trajectory-derived representation	Conformational geometry, trajectory-level moments, and learned physical proxies	Improved transfer in selected settings, particularly when combined with applicability-domain descriptors	Performance remains dependent on the downstream model and the type of chemical-distribution shift

3.6.1 Global Descriptors Capture Dominant Trends but Miss Chemical Context

Using the 116 processed descriptors, classical regressors were trained on each dataset. Support Vector Regression and XGBoost performed well in internal cross-validation, with FreeSolv yielding the lowest internal errors. Gini impurity-based XGBoost feature-importance scoring ranked TPSA highest, accounting for 54% of the total importance score, followed by the number of hydrogen-bond donors at 12%. This ranking is consistent with the expectation that polar surface area and hydrogen bonding are informative for hydration, but it should be interpreted as a model-specific importance score rather than a decomposition of the solvation free energy.

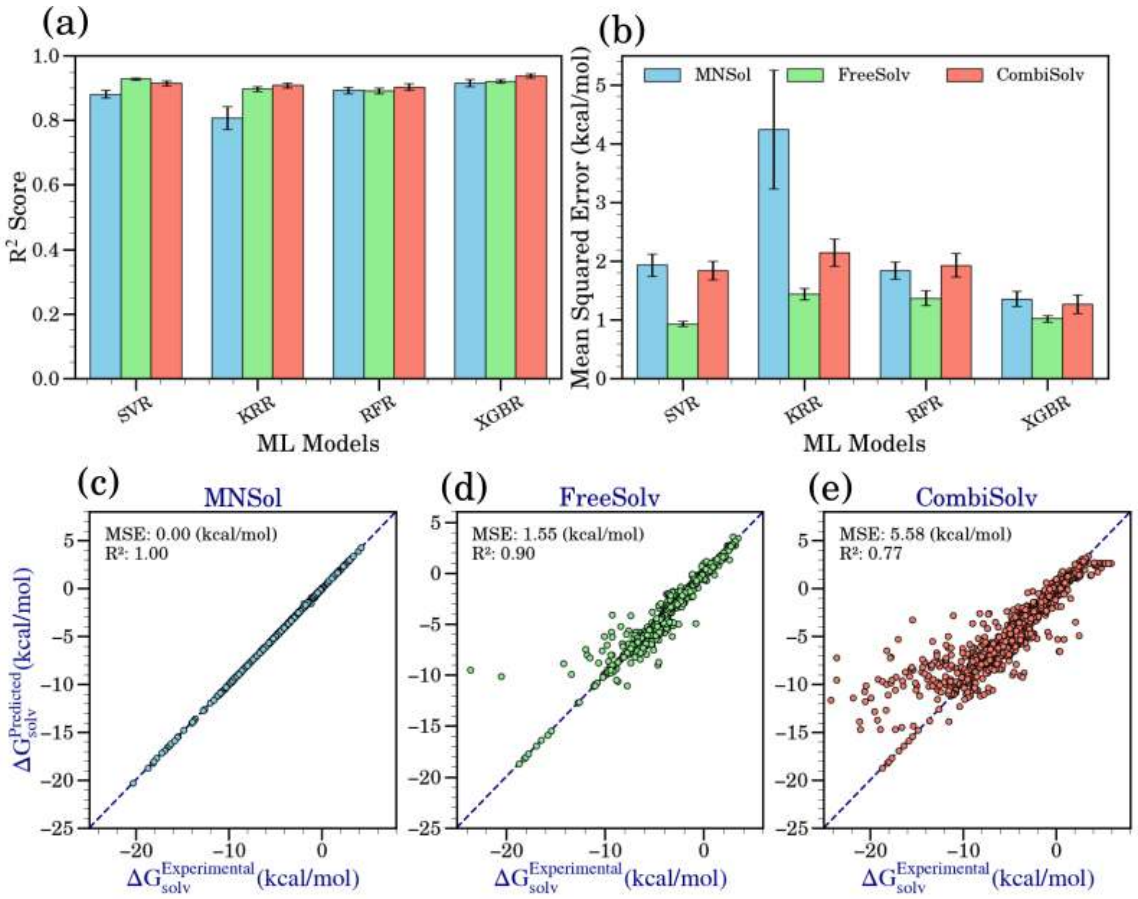

Figure 3.4: Descriptor-based model performance. (A,B) R^2 and MSE values for different machine-learning models across the datasets; MSE is reported in $(\text{kcal mol}^{-1})^2$. (C-E) Predicted versus experimental ΔG_{solv} values for transfer tests in which an XGBoost model trained on MNSol is evaluated on other datasets.

Figure 3.4 summarizes the descriptor benchmark and the corresponding MNSol-trained transfer tests. These transfer tests revealed limits in descriptor generalization. An XGBoost model trained on MNSol did not generalize well to FreeSolv and CombiSolv, showing large errors for compounds outside the training range. Representation bias

arises because global descriptors do not fully encode local atom environments or solvent accessibility. Physics-based features such as polar surface area, hydrogen-bond counts, and simplified electrostatic or cavitation proxies can improve transferability **[YadavPhysicsBased2025]**, but only when they capture the relevant physical failure mode. This motivates local structural representations, because a descriptor vector can identify that a molecule is polar but cannot always determine where that polarity sits in the molecular scaffold.

3.6.2 Fingerprints Resolve Local Motifs but Not Thermodynamic Scaling

Molecular fingerprints were evaluated to capture local atomic environments missed by global descriptors. MACCS keys and PubChem fingerprints outperformed hashed fingerprints using classical models. For example, XGBoost with MACCS keys achieved an MSE of 1.7 $(\text{kcal mol}^{-1})^2$ on CombiSolv, compared with 2.2 $(\text{kcal mol}^{-1})^2$ using Morgan fingerprints.

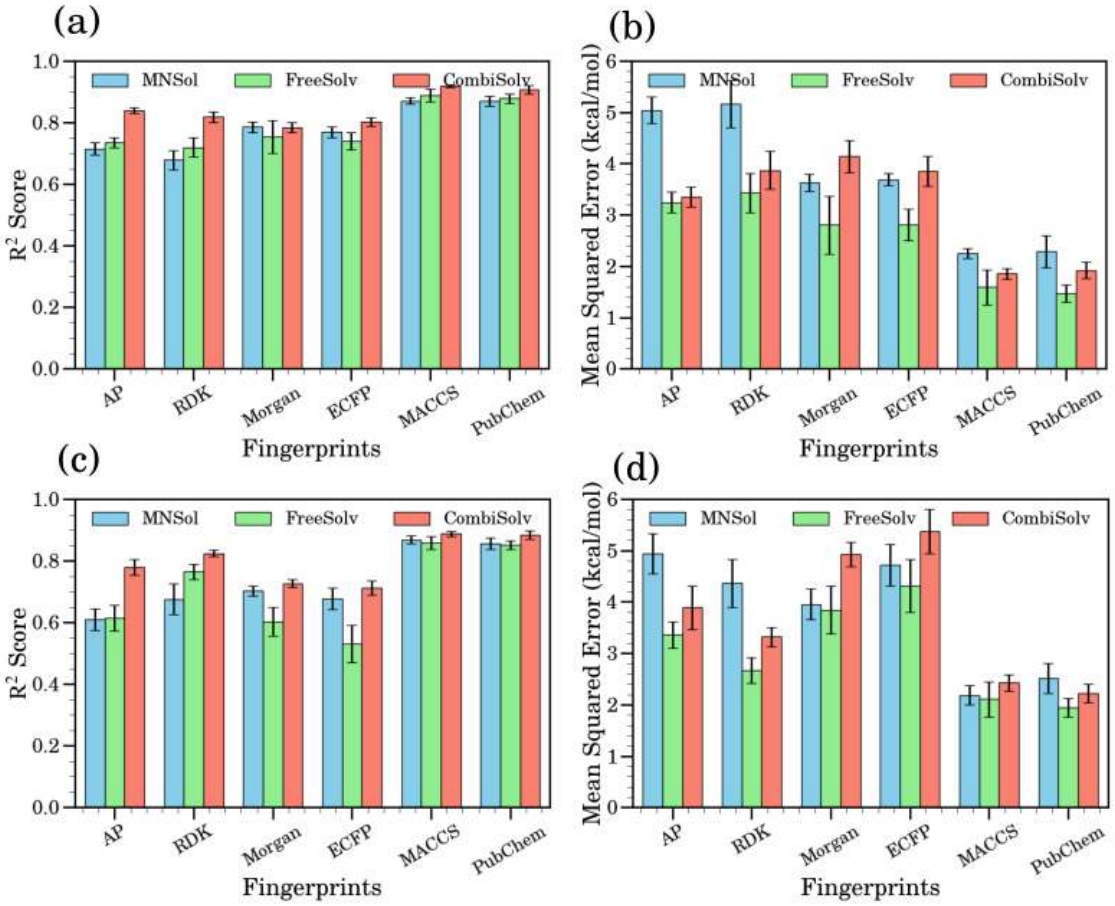

Figure 3.5: Comparison of molecular fingerprints using XGBoost and Random Forest Regression. The upper panels show XGBoost performance and the lower panels show Random Forest performance. Bar plots report R^2 and MSE values across datasets, with MSE in $(\text{kcal mol}^{-1})^2$ and standard error bars from five-fold cross-validation.

As shown in Fig. 3.5, the weaker performance of hashed fingerprints is attributed to bit collisions. Because many subgraphs are mapped to a fixed-length vector, chemically distinct fragments can activate the same bit. Structural keys reduce this ambiguity because each bit corresponds to a predefined fragment. To test whether deep networks could better use sparse hashed fingerprints, we trained a multilayer perceptron regressor. The MLPR reduced the MSE of Morgan fingerprints on CombiSolv from 2.2 to 1.1 $(\text{kcal mol}^{-1})^2$.

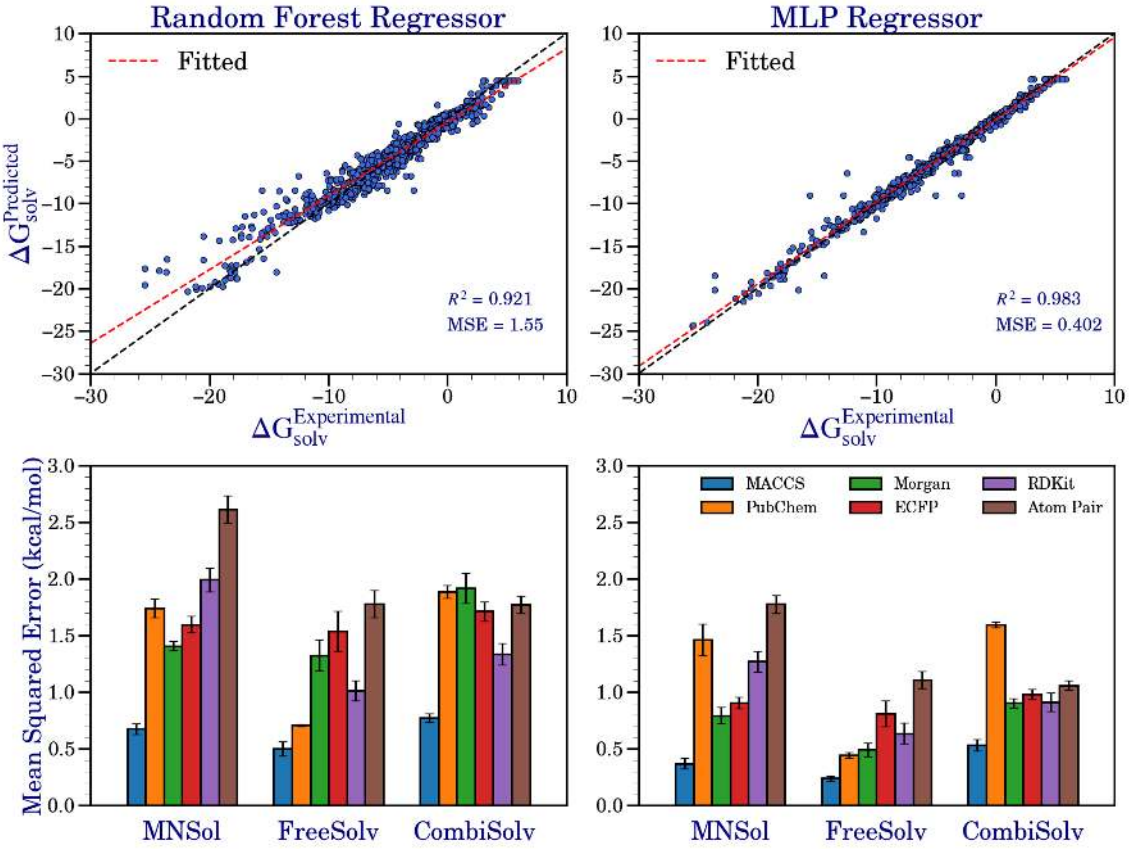

Figure 3.6: Comparison between Random Forest Regression and the multilayer perceptron regressor. (A,B) Predicted versus experimental solvation free energies on CombiSolv using Morgan fingerprints. (C,D) MSE comparison across datasets and fingerprint types, reported in $(\text{kcal mol}^{-1})^2$.

Figure 3.6 shows that the multilayer perceptron extracts more useful signal from Morgan fingerprints than the tree models in this test. Despite these improvements, fingerprint-based models still failed in transfer tests. For chemical classes absent from the training set, errors exceeded 10 kcal/mol. This confirms that fingerprints capture local structure but do not necessarily resolve the physical scaling of thermodynamic properties, such as the relation between hydrophobic surface area and cavity-formation free energy. The remaining gap motivates atomistic graph models and trajectory-derived representations, which can in principle encode molecular size, connectivity, and conformational structure more directly than fixed fragment vectors.

3.6.3 Graph Models Add Atomistic Detail but Do Not Guarantee Extrapolation

CIGIN was evaluated as an atom-level solute-solvent graph model and included as the graph-based baseline in Table 3.1. It achieved strong diagonal performance, but its transfer error increased sharply under distribution shift: when trained on MNSol it generalized reasonably to FreeSolv (MSE = 1.80 $(\text{kcal mol}^{-1})^2$) but deteriorated on CombiSolv (MSE = 5.48 $(\text{kcal mol}^{-1})^2$). Figure 3.7 details this transferability test, including the long-alkane error trend and the merged-dataset validation. The distinct CIGIN failure was systematic rather than random: for long-chain alkanes, absolute error increased with carbon-chain length, indicating that the model had not learned the cavitational cost of creating large hydrophobic cavities from a training set that lacked those molecules. This graph-model result closes the descriptor/fingerprint sequence: adding atomistic topology helps, but topology alone still does not guarantee physically reliable extrapolation.

The persistence of alkane and macrocycle errors motivates the final step in the chapter: learned trajectory-derived representations that expose three-dimensional conformational and physical information, followed by explicit applicability-domain descriptors for the chemotypes that remain difficult.

3.6.4 Trajectory-Derived Features Improve Transfer but Do Not Remove Chemotype-Specific Failures

The same transferability question was revisited with the trajectory-derived representation augmented with physically interpretable outlier descriptors. This analysis separates two questions: whether learned embeddings improve average prediction accuracy, and whether they correctly handle rare chemotypes that violate simple descriptor-property correlations. The outlier-aware descriptors were designed to close the loop opened by the earlier results: long-chain alkane and macrocycle failures reflect missing cavity and shape information, while descriptor and fingerprint failures for polar aromatics and rare linkers reflect misleading TPSA/H-bond counts and weak chemotype support. Ensemble spread, where used diagnostically, is interpreted only as empirical model disagreement and not as calibrated physical uncertainty.

Six controlled models were compared so that the contribution of the learned representation and the outlier-aware descriptors could be separated. The *QSAR baseline* is a Ridge model trained on six two-dimensional descriptors. The *trajectory-informed residual model* learns the residual error of this baseline using the trajectory-derived representation. Its outlier-aware counterpart receives the same trajectory-derived inputs together with the additional applicability-domain descriptors. To determine whether those descriptors are useful without trajectory-derived features, an *outlier-aware*

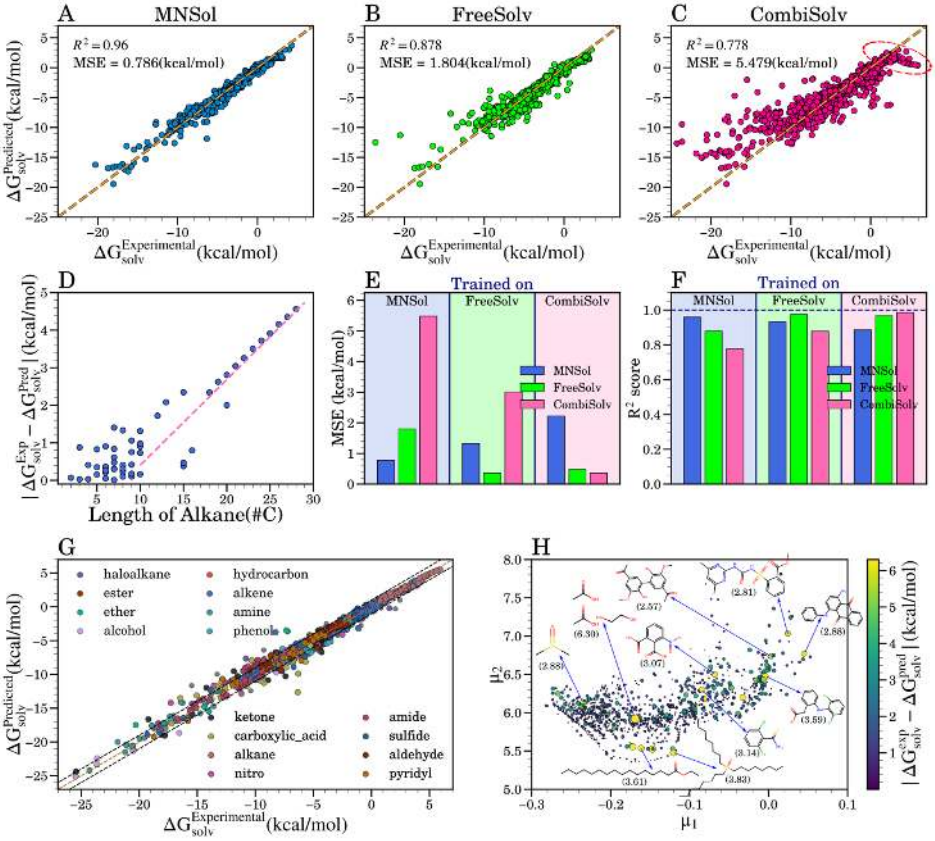

Figure 3.7: Transferability test for CIGIN. (A–C) Predicted versus experimental aqueous solvation free energies for a model trained on MNSol and evaluated on MNSol, FreeSolv, and CombiSolv, respectively. Performance is reported using R^2 scores and MSE values, with MSE in $(\text{kcal mol}^{-1})^2$, and the decline from MNSol to the external datasets shows the effect of novel solutes under chemical-distribution shift. (D) Absolute prediction error for alkane molecules from CombiSolv as a function of carbon-chain length, showing systematically larger errors for longer alkanes; these molecules are highlighted by the dashed circle in panel C. (E,F) Cross-dataset transferability summary, where CIGIN is trained on one dataset and tested on each of the three datasets, with performance shown by MSE and R^2, respectively. (G) Predicted versus experimental ΔG_{solv} for the merged dataset under k-fold cross-validation, with points colored by functional-group classification. (H) Representative high-error molecules projected into the GCN-VAE latent space, illustrating chemically localized failure motifs that remain difficult for the graph representation.

descriptor residual model was trained using descriptor information alone. Finally, the *direct trajectory-informed model* predicts ΔG_{solv} without a QSAR baseline, either with or without the outlier-aware descriptors. These names describe the scientific role of each model rather than the internal implementation used in the code.

Table 3.2: Transfer performance of the controlled trajectory-informed models. The CombiSolv test set contains 254 held-out molecules after leakage-aware filtering; MNSol is interpreted as a 65-molecule external diagnostic stress test. MAE values are reported in kcal mol^{-1} with nonparametric bootstrap 95% confidence intervals estimated from the molecule-level prediction errors.

Prediction strategy	**CombiSolv test MAE (95% CI)**	**MNSol stress-test MAE (95% CI)**
QSAR baseline	2.122 [1.906, 2.372]	2.873 [2.388, 3.530]
Trajectory-informed residual model	1.288 [1.127, 1.474]	2.027 [1.544, 2.647]
Outlier-aware trajectory-informed residual model	**1.131 [0.996, 1.288]**	2.019 [1.534, 2.636]
Outlier-aware descriptor residual model	1.484 [1.318, 1.670]	1.806 [1.387, 2.308]
Direct trajectory-informed model	2.032 [1.762, 2.330]	1.925 [1.504, 2.410]
Outlier-aware direct trajectory-informed model	1.555 [1.333, 1.784]	**1.662 [1.288, 2.062]**

Table 3.2 compares these controlled variants. The two external datasets tell different but complementary stories. On the held-out CombiSolv test set, the trajectory-informed residual model reduced the QSAR-baseline MAE from 2.122 to 1.288 kcal mol^{-1}. Adding the outlier-aware descriptors lowered the error further to 1.131 kcal mol^{-1}, giving the lowest observed CombiSolv test MAE. Paired bootstrap differences support this improvement relative to the trajectory-informed residual model (MAE difference = 0.160 kcal mol^{-1}, 95% CI [0.040, 0.287]) and the outlier-aware descriptor residual model (MAE difference = 0.355 kcal mol^{-1}, 95% CI [0.246, 0.465]). The same residual-model augmentation had almost no effect on MNSol: the MAE changed only from 2.027 to 2.019 kcal mol^{-1}, and the paired difference interval includes zero. The outlier-aware direct trajectory-informed model gave the lowest observed MNSol MAE, 1.662 kcal mol^{-1}, but its paired differences relative to the other top MNSol variants were not clearly separated from zero. Thus, the additional chemistry descriptors are useful, but their benefit depends on the prediction formulation and the type of distribution shift. The residual strategy is best supported on CombiSolv, whereas MNSol should be interpreted as a diagnostic stress test with overlapping uncertainty among the leading models.

The largest MNSol errors were not explained by formal charge. The stress-test molecules were formally neutral, but several occupied chemically rare or physically ambiguous regions: anthraquinones, sulfonylurea herbicides, and an organosilicon

molecule. These cases expose a common failure mode in which the model over-interprets polar group counts as solvent-accessible hydration benefit. In anthraquinones, polar groups are embedded in rigid planar aromatic scaffolds; in sulfonylureas, neutral-form representations may not match aqueous ionization behavior; and organosilicon chemistry lies outside the supported element domain.

The 30 outlier-aware descriptors can be grouped into five categories: size and surface accessibility (molecular weight, Labute ASA, TPSA/ASA), polarity and hydrogen bonding (TPSA, HBD/HBA, HBD/ASA, HBA/ASA), aromatic planarity (aromatic fraction, fused aromatic rings, F_{sp^3}), rare chemotype flags (anthraquinone, quinone, sulfonylurea, sulfonamide, carboxylic acid, silicon), and applicability-domain flags (unsupported elements, high planar polarity, high heteroatom density, and likely ionizable neutral form). These features improve interpretability because they identify when standard polarity descriptors may be misleading.

3.7 Conclusion

This chapter examined transferability as a property of the complete prediction pipeline rather than of the regression algorithm alone. The progression from descriptors to fingerprints, molecular graphs, and trajectory-derived embeddings revealed a consistent pattern. Global descriptors capture dominant trends in polarity and hydrogen bonding but lose the chemical context in which those features occur. Fingerprints restore local structural information, yet fragment counts do not describe how solvation free energy scales with molecular size, exposed surface, or conformational organization. Molecular graphs add atom-level connectivity and explicit solute–solvent interactions, but they still extrapolate poorly when the target molecule lies outside the size or chemotype range represented during training.

The trajectory-derived representation addresses part of this gap by encoding conformational geometry, energetic patterns, and trajectory-level physical proxies. Its value is clearest on the held-out CombiSolv test set, where the trajectory-informed residual model substantially improved upon the descriptor-based QSAR baseline and benefited further from outlier-aware descriptors. The MNSol stress test provides an important qualification: the same residual formulation did not improve appreciably after descriptor augmentation, whereas the best MNSol result was obtained by direct prediction from the trajectory-derived representation together with the outlier-aware features.

These results show that trajectory-derived information can improve external prediction, but it does not constitute a universally transferable representation. Its effectiveness depends on the downstream prediction strategy, the chemical composition of the target dataset, and whether the model is given information about its applicability domain.

The outlier analysis explains why. Several large errors arise not from formal charge alone, but from a mismatch between simple structural indicators and the physical environment of the functional groups. Polar atoms may be shielded by rigid aromatic scaffolds, nominally neutral molecules may be ionizable under experimental conditions, and unsupported elements may place a molecule outside the learned domain altogether. Exposing these conditions to the model improves both prediction and interpretation, because the model can distinguish ordinary polarity from polarity that is inaccessible, ambiguous, or chemically unfamiliar.

The central conclusion is therefore that transferability emerges from the interaction of four factors: the information contained in the molecular representation, the diversity of the training data, the downstream prediction strategy, and an explicit description of the applicability domain. Strong performance on a random split is insufficient evidence of transfer. A model should be called transferable only after it has been evaluated on a chemically independent test set, its dominant failure modes have been identified, and

the limits of its supported chemical domain have been reported. By this standard, none of the representations considered here is universally transferable, although trajectory-derived learning combined with physically motivated domain descriptors provides a more reliable route toward that goal.

The broader representation-learning strategy is extended in Chapter 4, where learned molecular descriptors are used to identify local crystal environments in condensed-phase water.

4

IceCoder: Self-Supervised Identification of Ice Polymorphs

4.1 Introduction

Chapter 3 examined how molecular representations (descriptors, fingerprints, and graphs) affect the transferable prediction of solvation free energies. That chapter addressed the *representation bottleneck*: the question of which encoding preserves the physics needed for accurate prediction across chemical space. This chapter shifts from molecular-level property prediction to structural identification in condensed-phase water, addressing the thesis's second bottleneck: the *identification bottleneck*, in which high-dimensional trajectory data must be converted into physically meaningful states, order parameters, or phases. Water exhibits one of the most complex phase diagrams among molecular liquids, featuring more than twenty confirmed crystalline ice polymorphs, multiple amorphous states, and several theoretically predicted phases [**Zheligovskaya2006**, **BartelsRausch2012**, **FuentesLandete**, **Zhu2020**]. This structural diversity presents a challenge for molecular dynamics simulations: during transitions such as nucleation, growth, melting, or polymorph conversion, phase identities must be resolved locally at the single-molecule level rather than through bulk averages. Here, we present *IceCoder*, a self-supervised representation-learning framework that uses Smooth Overlap of Atomic Positions (SOAP) descriptors and a Variational Autoencoder (VAE) to learn a continuous, low-dimensional order-parameter space for ice environments, followed by supervised phase assignment in that latent space.

4.2 Motivation

Hexagonal ice (ice-Ih) is the most prevalent crystalline form of water at ambient pressures, but many polymorphs exist across a range of temperature and pressure regimes [**Baker2019**, **Murray2006**, **Kamb1964**, **Hasted1969**, **Bauer2008**, **Kuhs1984**, **Buffett2004**, **Buchanan2005**, **Moon2003**, **B500373C**, **Johari2007**, **Stroeve2014**, **serreze2003record**]. Some of these phases are stable in planetary interiors, others occur in nanoconfined geometries, and many appear as metastable intermediates during high-pressure

crystallizations [**Hansen2021**, **Chaplin2019**, **Bore2022**, **Hirata2017**]. In simulations, these phases often emerge as small, transient clusters surrounded by liquid water. A local structural classifier must therefore distinguish subtle geometric features while remaining robust to thermal noise. Figure 4.1 illustrates the structural diversity of the phases considered here and motivates the use of local environment descriptors rather than bulk structural averages. Traditional order parameters rely on physical intuition.

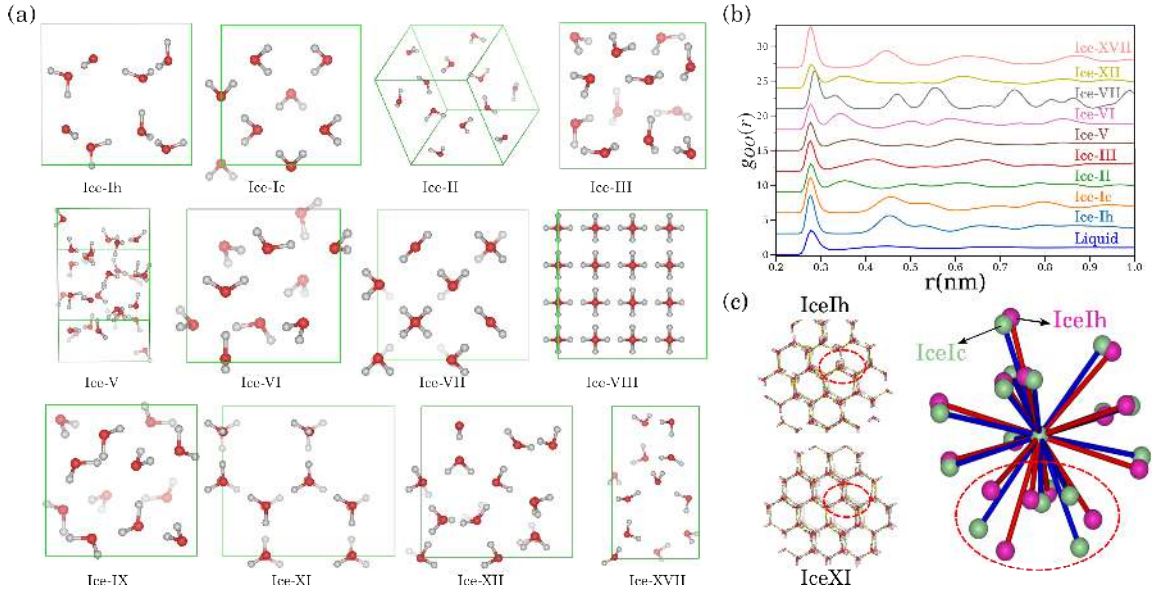

Figure 4.1: Structural diversity of ice phases used in this study. Representative unit cells, oxygen-oxygen radial distribution functions, and local-neighborhood comparisons illustrate why closely related phases can be difficult to distinguish using simple geometric criteria.

The structure of water has been analyzed using tetrahedrality, Steinhardt bond-order parameters, local entropy and four-body correlations, which have been widely used for this purpose [**Chau1998**, **Giovambattista2005**, **Yan2007**, **Zimmermann2017**, **Errington2001**, **Piaggi2017**, **Lechner2008**, **Reinhardt2012**, **Mickel2013**, **Crippa2023**, **Rodger1996**, **Choudhary2018**, **Choudhary2020**, **Shiratani1998**, **Wikfeldt2011**]. For example, the CHILL and CHILL+ algorithms classify hexagonal ice, cubic ice, clathrate structures and liquid water using local bond-order parameters [**Moore2010**, **Nguyen2014**]. These techniques are both fast and interpretable, but limited to a small set of low-pressure phases, and not easily extendable to the full spectrum of high-pressure polymorphs. Supervised machine learning algorithms, including deep neural networks and convolutional classifiers have made advances in phase classification [**Geiger2013**, **Fulford2019**, **DeFever2019**, **Kim2020**]. However, these methods are black-box classifiers and need pre-specified training labels. Conversely, the *IceCoder* framework uses self-supervised learning to build a continuous two-dimensional latent manifold, and then uses a supervised SVM only to assign phase labels within that learned plane. This latent space can act as an interpretable order-parameter plane,

which allows visualization, classification, and transition tracking without requiring ground-truth phase labels during VAE training.

Classical order parameters

4.2.1 Selection of Baseline Order Parameters

We first benchmarked several popular local order parameters to create a baseline. $Q(i)$ is the orientational tetrahedral order parameter for oxygen atom i and describes how similar the four nearest neighbors of oxygen atom i are to a regular tetrahedral geometry:

$$Q(i) = 1 - \frac{3}{8} \sum_{j=1}^{3} \sum_{k=j+1}^{4} \left(\cos \phi_{jk} + \frac{1}{3} \right)^2, \tag{4.1}$$

where ϕ_{jk} is the angle between the central oxygen and the neighbors j and k [**Errington2001**, **Jedlovszky2008**]. In a perfect tetrahedral arrangement $Q = 1$, while in disordered environments it is lower. Steinhardt bond-order parameters describe the angular distribution of neighboring atoms using spherical harmonics [**Plimpton1995**, **Reinhardt2012**]. For a central atom i, the local parameter $q_l(i)$ of integer rank l is:

$$q_l(i) = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |q_{lm}(i)|^2}, \tag{4.2}$$

where the orientational vector $q_{lm}(i)$ averages the spherical harmonics Y_{lm} over the set of $N_b(i)$ nearest neighbors:

$$q_{lm}(i) = \frac{1}{N_b(i)} \sum_{j=1}^{N_b(i)} Y_{lm}(\mathbf{r}_{ij}). \tag{4.3}$$

To improve robustness against thermal fluctuations, we use the locally averaged Steinhardt parameter $\bar{q}_l(i)$, which averages the q_{lm} vectors over the central atom and its first coordination shell:

$$\bar{q}_{lm}(i) = \frac{1}{N_b(i)+1} \left(q_{lm}(i) + \sum_{j=1}^{N_b(i)} q_{lm}(j) \right). \tag{4.4}$$

We also evaluated local entropy fingerprints. The two-body excess entropy S_2 is defined through the radial distribution function $g(r)$:

$$S_2 = -2\pi\rho k_B \int_0^\infty \left[g(r) \ln g(r) - g(r) + 1 \right] r^2 \, dr, \tag{4.5}$$

where ρ is the bulk number density. This global property is localized by computing a smoothed local radial distribution function around atom i [**Piaggi2017**, **Saija2003**, **Truskett2000**, **Baranyai1989**, **Nettleton1958**].

4.2.2 Baseline Performance and Limitations

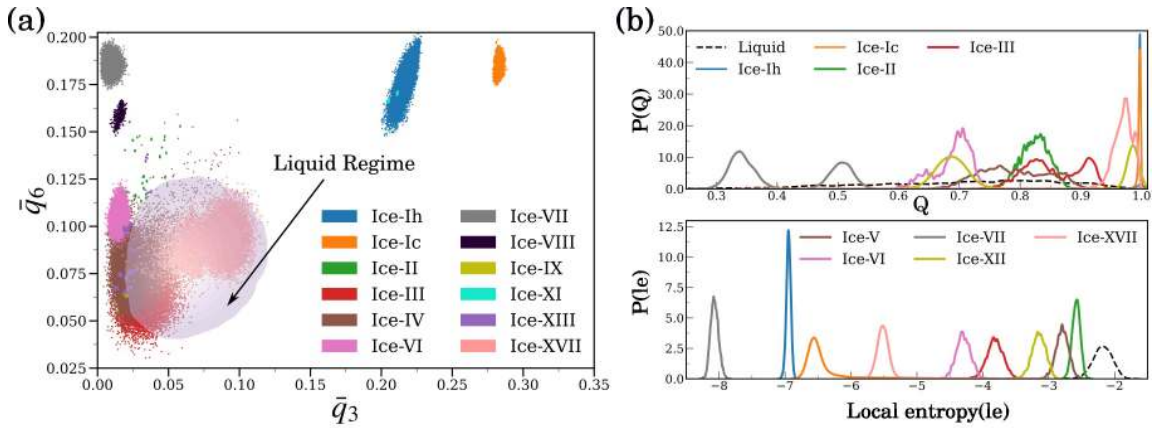

Figure 4.2: Classical order-parameter projections for multiple ice phases. The joint $\bar{q}_3$-$\bar{q}_6$ space separates some phases but leaves substantial overlap among others. One-dimensional distributions of tetrahedrality and local entropy show a similar limitation when many phases are considered simultaneously.

As shown in Fig. 4.2, these parameters resolve low-pressure phases like ice-Ih, ice-Ic, and liquid water, but they show significant overlap for high-pressure polymorphs (such as ice-II, V, VI, and XII). The thermal fluctuations of MD simulations expand the coordinate distributions, causing different phases to merge in the $\bar{q}_3$–$\bar{q}_6$ projection. This overlap highlights the need for a higher-dimensional descriptor and a non-linear projection model, the approach taken by *IceCoder* in the following section.

4.3 IceCoder Framework

4.3.1 SOAP Representation

To capture local structure without introducing coordinate system dependencies, we use the SOAP descriptor [**De2016**, **Maksimov2020**, **Himanen2020**]. SOAP represents the local neighbor density around each atom in a rotationally invariant form. Because the framework uses only oxygen atom positions (the heavy atoms of the water molecule), the descriptor is efficient and insensitive to the rapid librational motion of hydrogen atoms. However, this choice also means that phases differing only in hydrogen ordering (such as ice-Ih/XI, ice-III/IX, ice-V/XIII, and ice-VII/VIII) share identical oxygen lattices and are expected to map to adjacent regions in any oxygen-only descriptor space, a point we return to in Section 4.7. The local neighbor density $\rho_i(\mathbf{r})$ around an oxygen

atom i is represented by placing Gaussians on the surrounding oxygen atoms j:

$$\rho_i(\mathbf{r}) = \sum_j \exp\left[-\frac{|\mathbf{r} - \mathbf{r}_{ij}|^2}{2\sigma^2}\right]. \tag{4.6}$$

This density is expanded in a basis of orthogonal radial functions $g_n(r)$ and spherical harmonics $Y_{lm}(\hat{\mathbf{r}})$:

$$\rho_i(\mathbf{r}) = \sum_{nlm} c^{(i)}_{nlm} g_n(r) Y_{lm}(\hat{\mathbf{r}}). \tag{4.7}$$

The rotational invariance is achieved by forming the power spectrum:

$$p^{(i)}_{nn'l} = \sqrt{\frac{8\pi^2}{2l+1}} \sum_m \left(c^{(i)}_{nlm}\right)^* c^{(i)}_{n'lm}. \tag{4.8}$$

For this work, the local environment cutoff radius was set to $r_c = 10$ Å. The expansion parameters were $l_{\max} = 6$, $n_{\max} = 8$, and the Gaussian width was $\sigma = 0.25$ Å, yielding a 952-dimensional feature vector $\mathbf{p}^{(i)}$ for each oxygen environment. The descriptors were generated using DScribe and MinMax normalized prior to VAE training [**Himanen2020**].

4.4 Variational Autoencoder

The VAE regularizes the low-dimensional projection by mapping the input descriptor x to a Gaussian distribution in latent space z parametrized by a mean vector $\boldsymbol{\mu}$ and a variance vector $\boldsymbol{\sigma}$. In practice, the model is trained by minimizing the negative Evidence Lower Bound (negative ELBO), written as the sum of a reconstruction loss and a Kullback–Leibler regularization term:

$$\mathcal{L}_{\text{VAE}} = -\sum_i \mathbb{E}_{z_i \sim q_\phi(z_i|x_i)} \left[\log p_\theta(x_i|z_i)\right] + \text{KL}\left(q_\phi(z_i|x_i) \,\|\, p(z_i)\right), \tag{4.9}$$

where $q_\phi(z|x)$ is the encoder, $p_\theta(x|z)$ the decoder, and $p(z) = \mathcal{N}(0, \mathbf{I})$ the prior. The encoder architecture consists of fully connected layers of sizes 1500, 600, 400, and 100 nodes, projecting to a two-dimensional latent bottleneck. The decoder mirrors this architecture symmetrically. Gaussian Error Linear Unit (GELU) activations are used throughout [**GELU**]. Hyperparameters (layer sizes, learning rate) were selected using Optuna [**Optuna2019**]. The KL coefficient was fixed to $\beta = 1$, and early stopping was used during training. The network was trained with Adam optimizer [**adam**] with a learning rate of 10^{-5} and batch size 128. The final architecture and training settings are summarized in Table 4.1. Figure 4.3 summarizes the complete *IceCoder* pipeline from local oxygen environments to SOAP descriptors and VAE latent coordinates.

Table 4.1: VAE architecture and training hyperparameters for *IceCoder*.

Component	**Detail**
Encoder layers	1500 – 600 – 400 – 100 – 2
Decoder layers	2 – 100 – 400 – 600 – 1500
Activation	GELU
Optimizer	Adam, learning rate 10^{-5}
Batch size	128
KL coefficient	$\beta = 1$
Regularization	Early stopping
Hyperparameter optimization	Optuna

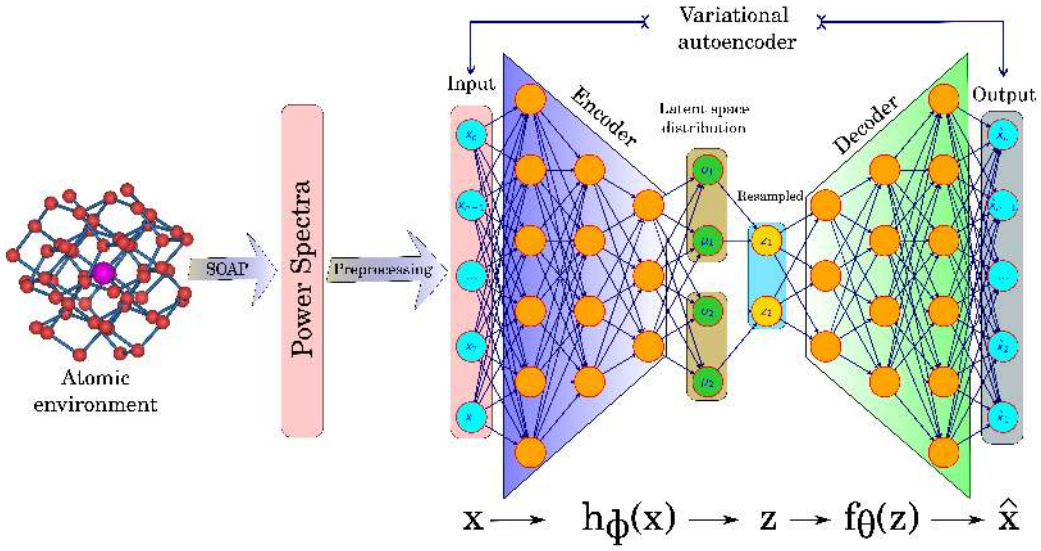

Figure 4.3: Schematic of the *IceCoder* workflow. Oxygen-centered local environments are converted into SOAP power spectra, normalized, and passed through a fully connected VAE. The two-dimensional bottleneck coordinates, μ_1 and μ_2, provide the learned structural order-parameter space.

4.5 Simulation Data

Initial ice configurations were generated using GenIce and minimized using steepest descent and conjugate gradient algorithms [**Matsumoto2017**]. We performed molecular dynamics simulations with the TIP4P/Ice water model [**Bore2022**, **Espinosa2014**, **HajiAkbari2015**, **Mochizuki2018**, **Louden2018**] which reproduces the experimental melting points and thermodynamic stabilities of water and ice polymorphs. The simulations were performed with GROMACS 2019.6 [**gmx1**, **Abraham2015**, **Berendsen1995**, **Abascal2005**, **Abascal2007**, **Jorgensen1988**, **Jorgensen1996**]. Electrostatic interactions were evaluated using PME with a 1.0 nm real-space cutoff, and Lennard-Jones interactions were truncated at 1.0 nm [**Essmann1995**]. The equations of motion were integrated with a 2 fs timestep, constraining bonds with LINCS [**Hess1997**, **DELHOMMELLE2001**]. Temperature and pressure were maintained using the velocity-rescale thermostat and

Parrinello-Rahman barostat, respectively [**Bussi2007**, **Parrinello1981**]. We performed

Table 4.2: Thermodynamic conditions and simulation cell dimensions used to generate homogeneous ice-phase trajectories.

Phase	Temperature (K)	Pressure (bar)	Simulation cell (nm^3)
Ice-Ih	230	1	$3.91 \times 3.67 \times 4.52$
Ice-Ic	230	1	$4.47 \times 4.47 \times 4.47$
Ice-II	230	3000	$3.91 \times 6.77 \times 3.65$
Ice-III	230	3000	$3.99 \times 3.99 \times 4.16$
Ice-V	230	6000	$3.68 \times 3.01 \times 5.85$
Ice-VI	230	8000	$3.71 \times 3.71 \times 3.42$
Ice-VII	230	50000	$4.68 \times 4.68 \times 4.68$
Ice-XII	250	1000	$4.79 \times 4.79 \times 4.67$
Ice-XVII	230	0.5	$6.23 \times 5.48 \times 5.96$

6 ns simulations for each homogeneous phase listed in Table 4.2. We prepared two heterogeneous validation systems:

1. A two-phase slab containing 2000 ice-Ih molecules and 2441 liquid water molecules, simulated for 100 ns at 265 K and 1 bar.

2. A three-phase system containing ice-Ih, ice-Ic, and liquid water, simulated for 500 ns.

4.6 Results and Discussion

4.6.1 Separation of Ice Phases in Latent Space

Having trained the SOAP–VAE model on minimized and finite-temperature configurations, we first assess whether the two-dimensional latent space separates distinct ice polymorphs. SOAP descriptors from energy-minimized configurations were projected into the 2D VAE space (μ_1, μ_2). Crystalline phases mapped to distinct, localized coordinates, as shown in Fig. 4.4. Hydrogen-ordered and hydrogen-disordered pairs, such as ice-Ih/XI, ice-III/IX, ice-V/XIII, and ice-VII/VIII, projected to adjacent regions. This proximity is expected, as these pairs share identical oxygen lattices, differing only in the proton arrangements. The hydrogen-ordered phases occupy smaller, more compact clusters in latent space, consistent with the hypothesis that their oxygen environments exhibit lower structural variability. The VAE projection maintained this separation when applied to finite-temperature MD trajectories. In contrast, projecting the same finite-temperature SOAP descriptors using linear Principal Component Analysis (PCA) resulted in significant overlap, with the liquid distribution obscuring several crystalline domains (Fig. 4.5, panel B). The non-linear layers of the VAE provide a useful way

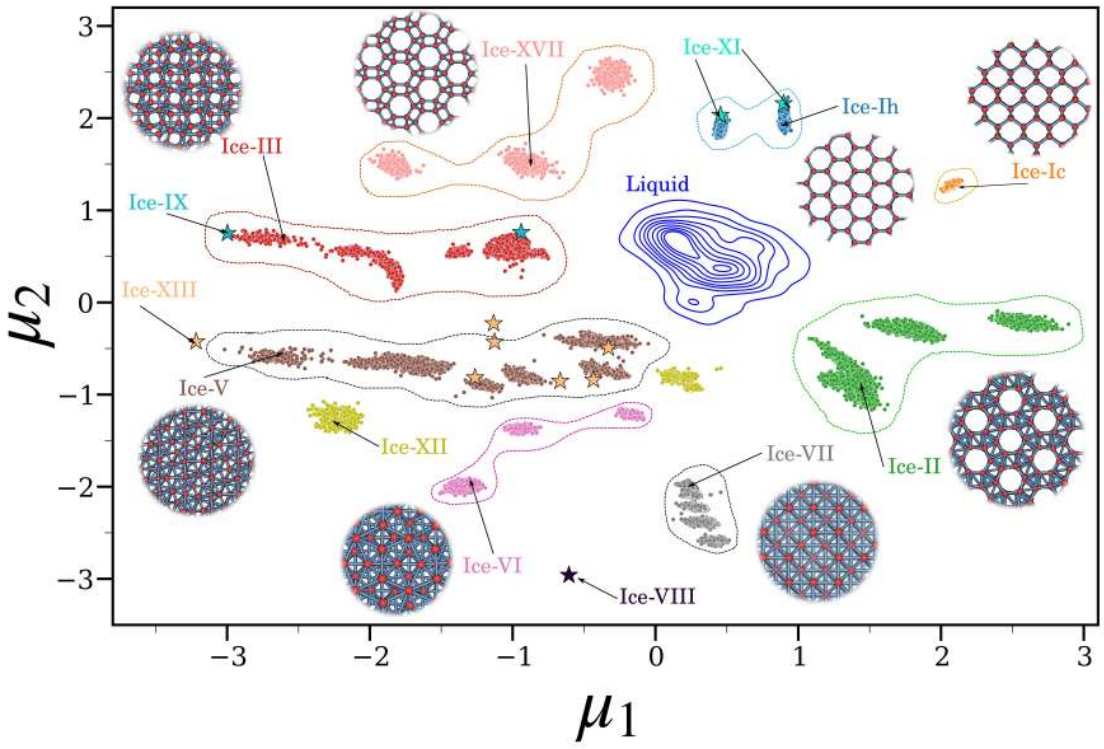

Figure 4.4: Projection of minimized ice structures into the two-dimensional VAE latent space. Most ice phases occupy distinct regions, while hydrogen-ordered counterparts appear near their corresponding disordered phases.

to resolve the geometric structure of the SOAP feature space under thermal fluctuations. PCA was chosen as the comparison baseline because it is the simplest linear projection and is widely used for trajectory analysis; non-linear alternatives such as t-SNE or UMAP would introduce additional hyperparameters and stochastic variation that complicate a direct assessment of whether the VAE's non-linear encoding alone improves phase separation. Figure 4.5 shows that the VAE separates many phases at finite temperature in this dataset. The following subsection examines whether the latent space also resolves structural variation *within* a single phase, a capability that is difficult for the classical order parameters considered here.

4.6.2 Intra-Phase Resolution

The VAE also resolved structural sub-states within a single phase. Projecting ice-VI configurations revealed three distinct subclusters. Applying DBSCAN clustering to this region [**ester1996density**] and analyzing the coordinate geometry suggested that these subclusters correspond to the orientation of water molecules relative to the two interpenetrating hydrogen-bonded networks of ice-VI. The VAE coordinates capture these localized coordinate variations, rather than acting only as categorical phase labels. This intra-phase resolution distinguishes the learned latent space from the classical order parameters tested here, which often collapse all configurations of a given phase

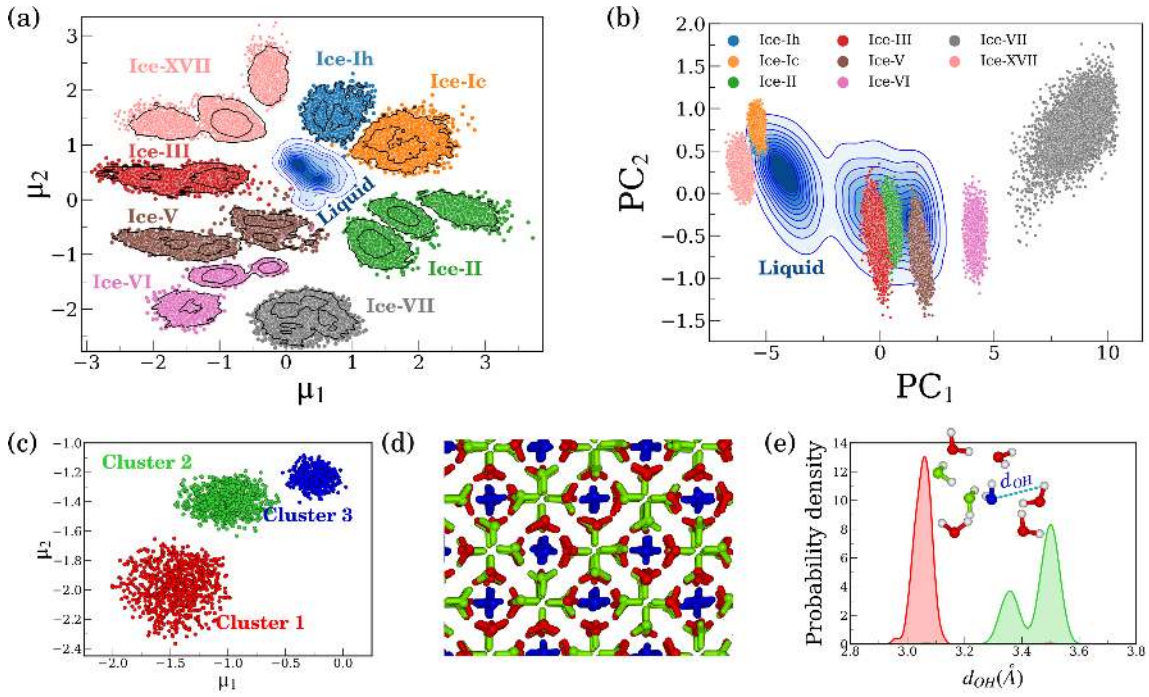

Figure 4.5: Finite-temperature ice and liquid environments in the learned latent space. (A) VAE projection of MD configurations. (B) PCA projection of the same SOAP vectors, showing stronger phase overlap. (C-E) Substructure within ice-VI, where latent-space clusters correspond to measurable differences in local geometry.

into a single distribution.

4.6.3 Automated Phase Assignment

To classify configurations automatically, we trained a Support Vector Machine (SVM) with a radial-basis-function (RBF) kernel on the VAE coordinates [**Cortes1995**, **scikit_learn**]. The SVM defines non-linear decision boundaries in the 2D plane (Fig. 4.6), mapping each projected molecule to a phase label. Qualitatively, the SVM decision boundaries aligned with the visually apparent cluster separations in the latent space. Misclassifications, where they occurred, were concentrated near the boundaries between adjacent phases in the latent plane, particularly between hydrogen-ordered and hydrogen-disordered pairs (e.g., ice-Ih/XI) and between phases with nearby latent coordinates (e.g., ice-VI and ice-VII at high pressure). A quantitative assessment of per-phase precision and recall, including train-test splitting and cross-validation, would require labeled data spanning the full range of thermal fluctuations and is left as a natural extension of the current work. By averaging the latent coordinates over all molecules in a simulation frame, we obtain global progress coordinates, $\langle \mu_1 \rangle$ and $\langle \mu_2 \rangle$. These coordinates monitor the macroscopic evolution of the system, while the individual molecule-wise labels track local structural changes. The preceding subsections established that the learned latent space separates ice phases and captures

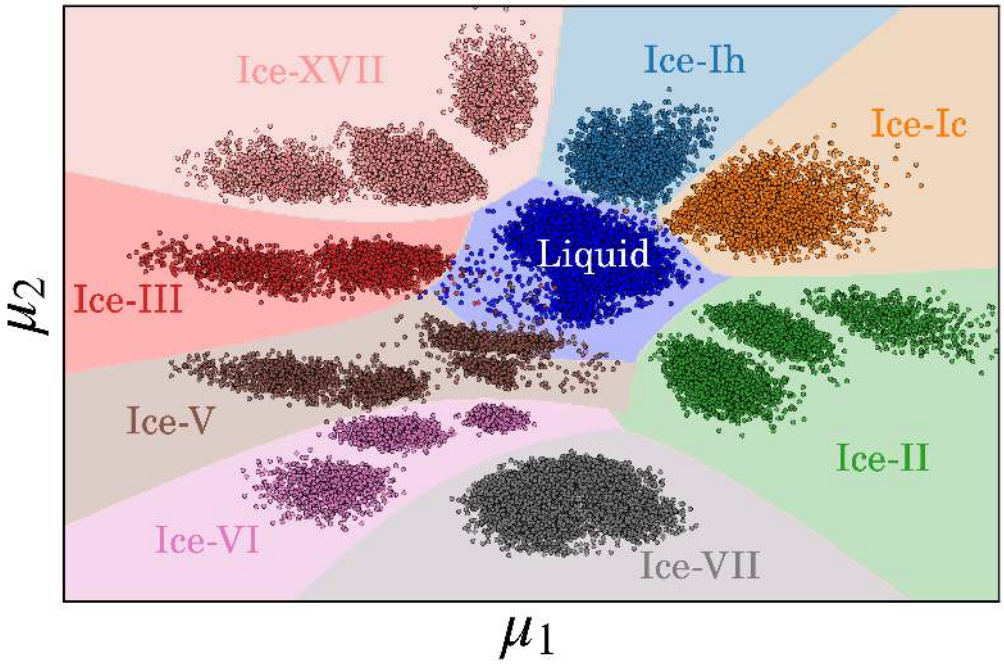

Figure 4.6: SVM decision regions in the $\mu_1 - \mu_2$ latent space. These boundaries convert the continuous VAE projection into local phase labels for molecular simulation trajectories.

intra-phase substructure. The next two subsections test whether this representation can track phase *evolution* in heterogeneous simulations that mimic experimentally relevant conditions.

4.6.4 Two-Phase Ice Growth

We tested the SOAP-VAE-SVM pipeline by monitoring the growth of ice-Ih from a liquid water interface. At the start of the simulation, the latent space showed separate liquid and ice-Ih populations. Over the 100 ns trajectory, the liquid population transferred into the ice-Ih basin. The frame-averaged coordinates ($\langle\mu_1\rangle, \langle\mu_2\rangle$) changed smoothly as the interface progressed. SVM phase counts agreed with CHILL+ classifications (Fig. 4.7), supporting the use of the VAE plane as a phase-conversion coordinate for this system. Having established that *IceCoder* tracks a single solid–liquid interface, we next evaluate whether it can simultaneously resolve three coexisting phases.

4.6.5 Three-Phase Conversion

We evaluated the classifier on a three-phase system containing coexisting ice-Ih, ice-Ic, and liquid water. During the trajectory, the liquid layer crystallized, and the system transitioned to a cubic-ice-dominated structure. The VAE coordinates resolved the three phases simultaneously. The frame-averaged coordinates tracked this transition, with one coordinate reflecting the reduction of hexagonal ice and the other tracking the

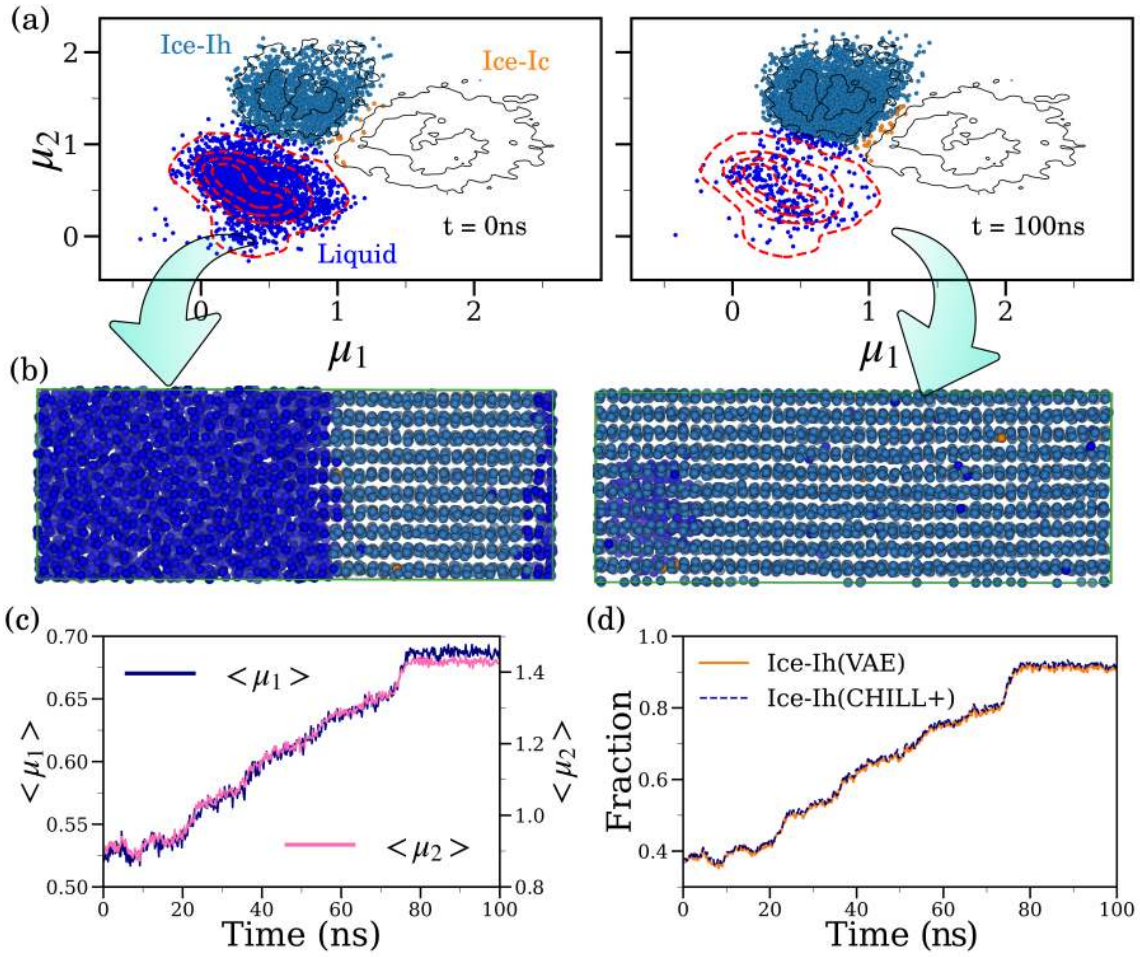

Figure 4.7: Hexagonal ice growth in a two-phase ice-Ih/liquid simulation. VAE projections, structure snapshots, frame-averaged latent coordinates, and comparison with CHILL+ show that *IceCoder* follows the conversion of liquid water into ice-Ih.

expansion of cubic ice. SVM phase counts matched CHILL+ classifications (Fig. 4.8), and the local nature of the descriptors allowed us to reconstruct the structural histories of individual molecules. Together, the two-phase and three-phase tests show that *IceCoder* not only separates static configurations but also follows the dynamical evolution of phase boundaries in heterogeneous systems. The agreement with CHILL+, a well-established order-parameter-based classifier, supports the latent-space assignments, while the VAE additionally resolves intra-phase substructure (ice-VI subclusters) and provides a continuous coordinate for each molecule, enabling the molecule-level trajectory reconstruction shown in the three-phase simulation.

4.7 Conclusions

This chapter presented the *IceCoder* SOAP-VAE framework for identifying ice polymorphs in molecular simulations. While classical order parameters struggle to resolve high-pressure crystalline phases under thermal fluctuations, combining SOAP descriptors

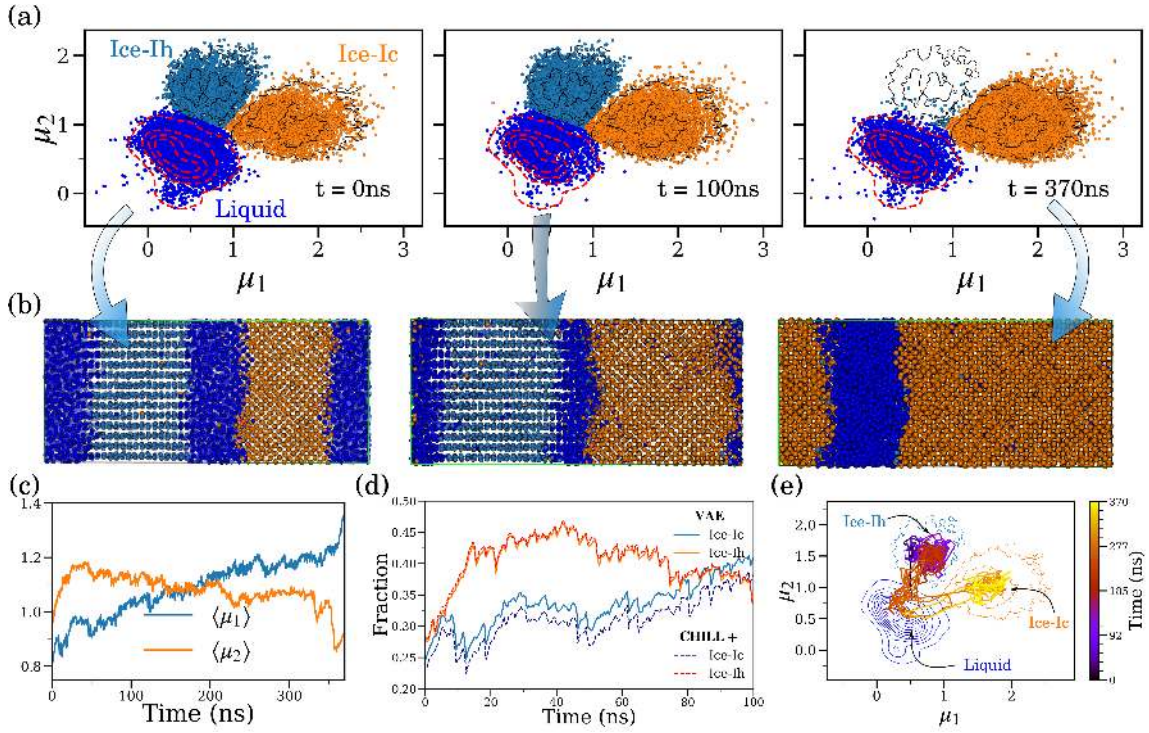

Figure 4.8: Phase evolution in a mixed ice-Ih/ice-Ic/liquid simulation. The VAE projection, structural snapshots, latent-coordinate time series, phase counts, and a single-molecule trajectory show the conversion toward cubic ice.

with a VAE yields a continuous, 2D latent space that separates nine ice phases and liquid water. The VAE latent space serves two roles:

1. Globally, frame-averaged coordinates act as order parameters to monitor phase transitions.

2. Locally, individual molecular environments are projected and classified using SVM decision boundaries.

Tests on slab-coexistence simulations showed agreement with CHILL+ while providing additional resolution of local structural sub-states. The main limitation is that hydrogen-ordered and hydrogen-disordered pairs (ice-Ih/XI, ice-III/IX, ice-V/XIII, ice-VII/VIII) map to the same region because the SOAP descriptor uses only oxygen positions. Since these phases share identical oxygen lattices and differ only in proton ordering, an oxygen-only descriptor cannot resolve them. Extending the framework to include hydrogen coordinates, for example by computing SOAP descriptors on both oxygen and hydrogen atoms or by augmenting the feature vector with orientational information, would enable the latent space to discriminate these phases. This extension is a natural direction for future work, particularly for simulations where proton ordering influences thermodynamic or kinetic properties.

Beyond static classification, the learned latent space provides a continuous, low-dimensional coordinate that could serve as a collective variable for enhanced sampling.

The μ_1–μ_2 plane separates physically distinct states and varies smoothly across phase transitions, two properties that make it suitable for biasing or adaptive sampling protocols. Chapter 5 develops this logic in the context of rare-event kinetics by combining pathway generation, learned representations, and path-resolved weighted ensemble simulations to estimate route-specific rate constants. While *IceCoder* addresses the *identification* bottleneck (what states are present), Chapter 5 addresses the *sampling* bottleneck (how to observe and quantify transitions between states), and the representational principles developed here, in which high-dimensional local descriptors are compressed into a learned low-dimensional space, reappear as a foundation for the path collective variables and latent adaptive sampling spaces of later chapters.

5

PathGennie and Path-Resolved Weighted Ensemble Kinetics: From Rapid Pathway Discovery to Quantitative Route-Specific Rate Constants

5.1 Introduction

Rare molecular events, including ligand dissociation, protein folding, conformational switching, and phase transformations, are central to biological function yet extraordinarily difficult to observe using conventional molecular dynamics (MD) simulations [**Copeland2006**, **HenzlerWildman2007**, **Shaw2010**]. The origin of this difficulty lies in a mismatch between the integration timestep, which is constrained to 1–2 fs to resolve the fastest bond vibrations [**Leach2001**], and the process timescales of interest, which frequently span microseconds to seconds. Consequently, simulations spend the overwhelming majority of their computational budget fluctuating within metastable basins while waiting for a productive thermal fluctuation that initiates a transition [**Chandler1987**]. Once initiated, the actual barrier-crossing event is typically brief (nanoseconds), but the preceding waiting time renders it effectively inaccessible.

Many crucial biomolecular processes, including ligand dissociation, conformational switching, and folding, frequently proceed through multiple competing transition channels. Although these channels lead to identical macroscopic outcomes, they can contribute very differently to the overall kinetics. Mutations, environmental perturbations, or changes in solvent conditions may alter not only the total transition rate but also the relative weight of individual pathways. Understanding how reactive flux is partitioned among competing channels is therefore essential for establishing a mechanistic picture of biomolecular function.

This *timescale problem* has motivated the development of numerous enhanced sampling strategies. CV-biasing methods, including umbrella sampling [**Kumar1992**], metadynamics [**Laio2002**, **Barducci2008**], and on-the-fly probability enhanced sampling (OPES) [**Invernizzi2020**], lower free-energy barriers along selected slow degrees of freedom but depend critically on the *a priori* identification of suitable reaction coordinates.

If important slow degrees of freedom are omitted, the resulting free-energy surfaces suffer from hysteresis and hidden barriers [**Rohrdanz2011**]. Path-based approaches including transition path sampling (TPS) [**Bolhuis2002**, **Dellago1998**, **Bolhuis2021**], forward flux sampling (FFS) [**Allen2009**], and the string method [**E2005**, **Elber2020**] focus computational effort directly on the reactive ensemble, but require either an initial transition path or a carefully defined set of progress interfaces. Markov state models and milestoning approaches [**Prinz2011**, **Faradjian2004**, **Votapka2017**, **Votapka2022**] provide systematic kinetic frameworks but depend on adequate configurational coverage of the transition region. The weighted ensemble (WE) framework offers a statistically rigorous alternative by propagating many weighted trajectories in parallel and periodically redistributing statistical effort through splitting and merging, thereby efficiently sampling low-probability transition regions while preserving unbiased kinetics [**Huber1996**, **Zuckerman2017**].

A well-recognised limitation of conventional WE simulations is the potentially long transient relaxation period before walkers successfully populate high-barrier transition regions. When all walkers are initialised exclusively within the reactant basin, significant simulation time may be spent simply diffusing away from the starting state before productive barrier-crossing trajectories emerge [**Bhatt2010**, **Miao2020**, **Ahn2021**]. Several studies have demonstrated that this bottleneck can be substantially alleviated by pre-populating bins along the reaction coordinate at the initial iteration, thereby providing immediate coverage of relevant regions of phase space. Furthermore, most kinetic analyses reduce the transition process to a single global rate constant. In many systems this simplification obscures important mechanistic heterogeneity: standard WE simulations along a single global coordinate can inadvertently suppress low-probability mechanistic branches through trajectory merging, making pathway-specific kinetic characterisation particularly challenging [**Zuckerman2017**, **Torrillo2021**, **Bose2025**].

Recent data-driven approaches have substantially improved our ability to identify and characterise mechanistically distinct pathways [**Rydzewski2019**, **Ray2024**, **Tnzel2024**, **Leung2025**]. Ray and Parrinello demonstrated that unbinding trajectories can be classified into mechanistically distinct families using machine-learned descriptors [**Ray2024**], and Elangovan, Chatterjee, and Ray extended this idea to drive enhanced sampling preferentially along selected mechanistic channels [**Elangovan2025**]. These methods establish which pathways exist and how to sample them efficiently, but stop short of assigning rigorous, pathway-specific kinetic observables. A complementary framework is required that couples pathway discovery directly to quantitative, pathway-resolved rate estimation.

This chapter describes such a framework, developed over two successive studies. The first introduced *PathGennie* [**Maity2025**], a direction-guided adaptive sampling strategy that generates diverse, physically plausible candidate transition pathways at

computational costs orders of magnitude below conventional MD. The second built upon PathGennie output to introduce a *path-resolved weighted ensemble* framework that clusters generated pathways using Dynamic Time Warping (DTW), refines them into smooth reference curves using neural networks, transforms the refined curves into path collective variables (PathCVs) [**Branduardi2007**], and performs independent WE simulations along each PathCV to resolve route-specific rate constants. Together, the two works provide a complete pipeline from rapid exploratory path generation to quantitative, mechanism-specific kinetics.

5.2 Background and Motivation

The central challenge in simulating activated processes is that straightforward MD is computationally efficient for local equilibrium sampling but exponentially inefficient for barrier crossing. Enhanced sampling methods attempt to solve this by either flattening the free-energy landscape (CV-based biasing), expanding the accessible ensemble (generalized ensembles), or selectively propagating successful trajectories (path sampling).

CV-based biasing methods, including umbrella sampling [**Kumar1992**], metadynamics [**Laio2002**], and OPES [**Invernizzi2020**], add external potentials to the system's Hamiltonian along a small set of slow degrees of freedom. While these methods can yield accurate free-energy profiles when the CVs are well chosen, they face two persistent difficulties. First, finding good coordinates for complex molecular systems usually requires a large amount of chemical intuition or post-hoc analysis [**Noe2017**]. Second, obtaining unbiased kinetic rates from biased trajectories relies on delicate reweighting procedures that can introduce statistical error, especially in the case of large bias or suboptimal coordinate [**Ray2022**, **Widmer2023**, **Salvalaglio2014**].

Methods that rely on paths, such as TPS [**Bolhuis2002**, **Bolhuis2021**], FFS [**Allen2009**] and the string method [**E2005**, **Elber2020**], concentrate the computational effort directly on the transition ensemble. TPS generates an ensemble of reactive trajectories through a Monte Carlo procedure in trajectory space, but requires an initial reactive path to seed the algorithm. FFS introduces a set of interfaces between reactant and product states and launches trial trajectories from each interface, yet its efficiency depends sensitively on interface placement. Both methods can struggle in high-dimensional systems with multiple competing pathways.

The weighted ensemble framework occupies a distinct niche. By partitioning trajectories into bins and maintaining a target number of walkers in each bin, WE enhances the sampling of rare states without altering the underlying dynamics [**Huber1996**, **Zuckerman2017**]. Because trajectories evolve under the unbiased Hamiltonian, rate constants can be estimated directly from the flux of probability into the product state.

However, WE efficiency depends critically on the initial distribution of walkers. When all walkers are initialized in the reactant basin, the algorithm must first discover the transition region before it can efficiently split and propagate weight across the barrier. This transient "discovery phase" can consume the majority of the computational budget, particularly in systems with complex, multi-channel transition pathways.

PathGennie was designed to eliminate this discovery phase while simultaneously revealing the mechanistic structure of the transition. The central concept is to focus computational effort on the transition region itself by generating many short trial trajectories and propagating only those that make progress along a progress coordinate. Because the equations of motion remain unbiased, the resulting trajectories are physically plausible, and the algorithm can rapidly cross barriers that would require microseconds or longer in conventional MD. The generated pathways then serve two purposes: (1) initial configurations for WE simulations, bypassing the transient phase, and (2) the raw material for pathway classification and PathCV construction.

5.3 The PathGennie Algorithm

PathGennie operates through an iterative cycle of short exploratory trials and conditional propagation. The algorithm is initialized from an equilibrated conformation in the source metastable basin. In each cycle, the algorithm executes a *sampler* segment of length τ_1 with randomised initial velocities. The endpoint of this trial is projected into a user-defined CV space and evaluated against a geometric acceptance criterion. Two modes of operation are supported:

- **Escape Mode:** The trial is accepted if it increases the distance from the source state A:
$$\|\xi(X_{\text{trial}}) - \xi(A)\| > \|\xi(X_{\text{start}}) - \xi(A)\|. \tag{5.1}$$
This mode is appropriate for unbinding and basin-escape transitions where the target state is not well defined or is structurally heterogeneous (e.g. the fully solvated unliganded protein).

- **Directed Mode:** The trial is accepted if it decreases the distance to a target state B:
$$\|\xi(X_{\text{trial}}) - \xi(B)\| < \|\xi(X_{\text{start}}) - \xi(B)\|. \tag{5.2}$$
This mode is used when both endpoints are known, such as in protein folding or conformational switching between two crystallographically resolved states.

If a trial satisfies the acceptance criterion, the algorithm executes a *runner* segment of length τ_2 to relax the configuration along the accepted direction. The final structure

of the runner segment becomes the starting point for the next cycle. If the trial is rejected, a new sampler is launched from the current position. To prevent trapping in entropic bottlenecks where no trial satisfies the criterion, the algorithm launches a fallback trajectory of length $\tau_1 + \tau_2$ after N_{trial} consecutive rejections.

The parameters τ_1, τ_2, and N_{trial} govern a fundamental trade-off between exploration and relaxation. Short sampler segments generate diverse trial directions, increasing the probability of finding a productive path over the barrier, while short runner segments preserve momentum along the productive direction, producing direct, high-energy crossing pathways. Conversely, longer runner segments allow the system to relax into local free-energy minima, generating pathways that more closely follow minimum-free-energy channels at the cost of slower overall progress. The parameter N_{trial} controls the persistence of the search before surrendering to a fallback trajectory. The code is flexible with respect to parameter choices: for the 3PTB trypsin system, $\tau_1 = 40$ fs, $\tau_2 = 100$ fs, and $N_{\text{trial}} = 50$ were used; for the imatinib systems, parameters in the range $\tau_1 = 10$–20 fs, $\tau_2 = 20$–50 fs, and $N_{\text{trial}} = 25$–50 were tested, and the relative pathway populations were found to be robust to this range of choices.

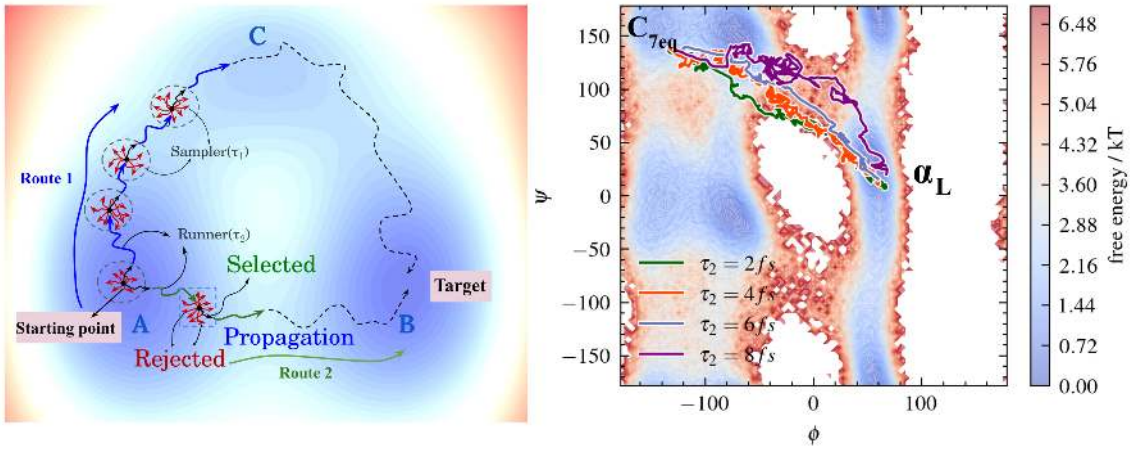

Figure 5.1: Schematic of the PathGennie sampling cycle and alanine dipeptide benchmark. (Left) The algorithm alternates between short sampler trials (blue) and conditional runner propagation (orange). (Right) Generated pathways for alanine dipeptide in the ϕ–ψ plane, illustrating how increasing the runner length τ_2 shifts trajectories from direct crossings toward minimum-free-energy routes.

As an initial benchmark, we applied PathGennie to alanine dipeptide in explicit solvent, targeting the transition from the $C_{7\text{eq}}$ basin to the α_L basin. Backbone ϕ and ψ dihedrals served as the CV space. Varying the runner length τ_2 from 2 to 8 ps (with $\tau_1 = 2$ ps and $N_{\text{trial}} = 15$) systematically shifted the character of the generated pathways: shorter runners produced direct, high-energy crossings, whereas longer runners allowed trajectories to relax toward the minimum-free-energy isocommittor surfaces (Fig. 5.1).

5.4 Simulation Systems and Computational Details

We validated PathGennie and the path-resolved WE workflow across a hierarchy of system complexity: analytical model potentials, small-molecule ligand unbinding from proteins, and fast-folding protein folding/unfolding transitions.

5.4.1 Force Fields and System Preparation

For alanine dipeptide and the mini-protein chignolin, we employed standard force fields (AMBER99SB-ILDN and OPLS-AA/L, respectively) with explicit TIP3P solvent [**LindorffLarsen2010**, **Kaminski2001**]. For Trp-cage and Protein G folding, simulation parameters were matched to the long-timescale reference trajectories from D. E. Shaw Research, enabling direct comparison with published free-energy surfaces [**LindorffLarsen2011**, **Piana2011**].

The Abl kinase–imatinib complex was built from the crystal structure PDB 1OPJ. System preparation used CHARMM-GUI [**Jo2008**, **Jo2014**, **Lee2015**]: the protein was described by the CHARMM36m force field [**Huang2016**] and imatinib parameters were generated with the CHARMM General Force Field (CGenFF) [**Vanommeslaeghe2009**, **Vanommeslaeghe2012**]. The complex was solvated in a cubic TIP3P box [**Jorgensen1983**] with a minimum protein-surface-to-boundary padding of 10 Å, and Na^+/Cl^- ions were added to reach a physiological ionic strength of 150 mM. The N368S mutant was prepared identically. We selected 1OPJ to maintain consistency with the infrequent-metadynamics study of Shekhar et al. [**Shekhar2022**] against which our rates are compared.

For benzene unbinding from T4 lysozyme L99A, we initialized from the bound crystal structure (PDB 3DMV) [**Kamenik2021**]. Benzene parameters were generated using LigParGen [**Dodda2017**], with the protein described by OPLS-AA/L and solvent by SPC [**Kaminski2001**, **Wu2006**].

For benzamidine–trypsin, atomic coordinates were taken from PDB 3PTB. Protonation states were assigned at pH 7.7 to reproduce prior experimental conditions [**Ray2021**, **Votapka2017**]. The protein was parameterised with AMBER ff14SB [**Maier2015**] and benzamidine with GAFF [**Wang2004**]. The system was solvated in a truncated octahedral box of TIP4P-Ew water [**Horn2004**] with eight Cl^- ions for neutrality, yielding approximately 23,000 atoms in total.

The host–guest complex OAMe-G2 was adopted from the PLUMED-NEST repository corresponding to the SAMPL7 challenge set [**Bonomi2019**, **Sun2020**]. Simulation protocols closely followed the reference setup for that benchmark to ensure comparability with prior studies.

All molecular dynamics simulations were performed with OpenMM 8.0 or GROMACS 2022.4 [**Eastman2017**, **Abraham2015**, **Pll2015**]. A Langevin middle integrator

maintained constant temperature (298 K for trypsin, 300 K for Abl kinase) with a friction coefficient of 5 ps^{-1}. Pressure was controlled at 1 atm using a Monte Carlo barostat. Integration timesteps of 2–4 fs were employed using hydrogen mass repartitioning [**Hopkins2015**].

5.4.2 Equilibration Protocol

Each system was first energy-minimised by steepest descent, then equilibrated for 1 ns under NVT conditions followed by 5 ns under NPT conditions at the target temperature and 1 bar. Harmonic positional restraints on protein backbone atoms and ligand heavy atoms were gradually released over the equilibration period. Long-range electrostatics were treated with particle mesh Ewald (PME) and hydrogen bonds were constrained throughout.

5.4.3 Collective Variable Construction for Ligand Unbinding

For ligand unbinding studies, the CV space was constructed using PCA of a synthetic conformational ensemble. Starting from the bound crystal structure, decoy ligand poses were generated by random rigid-body translations and rotations while keeping the protein fixed. Structures with severe steric clashes were discarded. For each retained pose, a feature vector was built from pairwise distances between ligand heavy atoms and nearby protein C_α atoms. PCA was applied to the resulting distance matrix, and the leading components explaining 95% of the cumulative variance were retained (4 components for benzene; 5 components for imatinib and benzamidine). For OAMe-G2, synthetic conformers were additionally constrained within a conical region extending outward from the host cavity, producing a more physically meaningful PCA manifold that captures both wet and dry unbinding channels. This data-driven approach captures dominant modes of ligand motion without requiring manual selection of specific interatomic distances. More expressive machine-learned collective variables such as VAMPnets or Deep-TICA [**Mardt2018**, **Bonati2021**, **Ribeiro2018**] can be substituted directly into this framework.

5.5 Robustness to an Imperfect Progress Variable

A persistent challenge in enhanced sampling is that optimal reaction coordinates are rarely known in advance. To test PathGennie's robustness to coordinate misspecification, we applied the algorithm to the two-dimensional Wolfe-Quapp potential, a standard test system featuring two competing saddle channels connecting two minima. We intentionally used only the x coordinate as the progress variable, leaving the orthogonal y coordinate completely unguided, even though successful barrier crossing requires correlated motion along both dimensions.

System	$(N_{\text{trial}}, \tau_1, \tau_2)$ [ps]	Reactive length [ps]	Total length [ps]	Wall time
Benzene unbinding	$(25, 0.01, 0.05)$	98 ± 22	228 ± 56	~2 min
Imatinib unbinding	$(25, 0.02, 0.03)$	184 ± 49	639 ± 224	~4 min
Trp-cage unfolding	$(15, 0.02, 0.08)$	150 ± 33	618 ± 139	~1.5 min
Protein G unfolding	$(15, 0.01, 0.09)$	176 ± 67	312 ± 128	~3 min
Trp-cage folding	$(50, 0.01, 0.04)$	336 ± 95	430 ± 122	~5 min
Protein G folding	$(50, 0.01, 0.04)$	612 ± 121	1593 ± 359	~15 min

Table 5.1: Representative PathGennie parameters and computational costs. Reactive length denotes the cumulative time along accepted productive segments; total length includes unproductive trials. Wall times correspond to single-trajectory costs on one NVIDIA RTX 2080 Ti GPU.

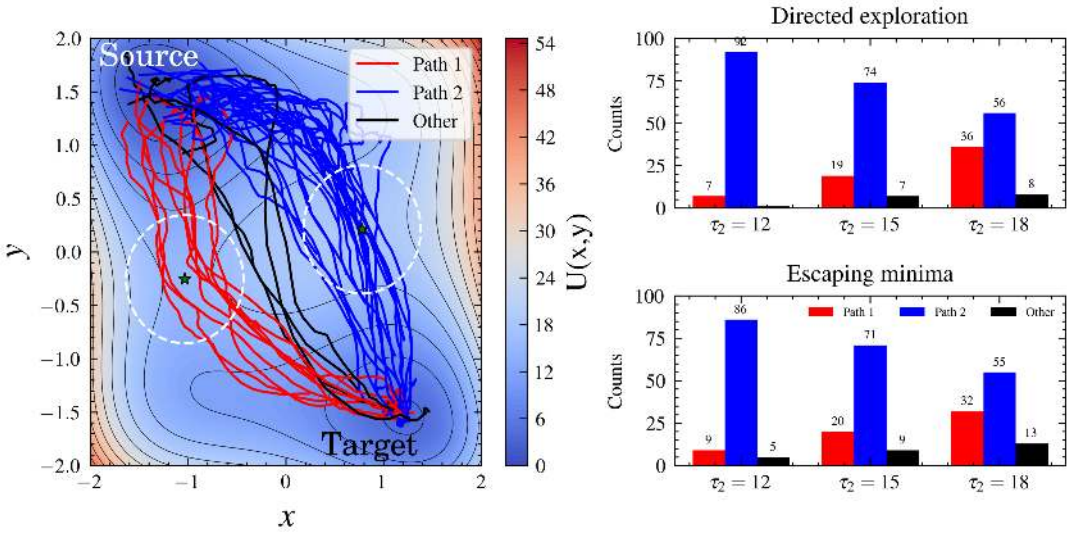

Figure 5.2: PathGennie trajectories on the Wolfe-Quapp potential with a suboptimal progress variable. Despite guiding only along x, the algorithm discovers transitions through both the upper and lower saddle channels. Increasing the runner length τ_2 shifts the path distribution toward the lower-energy channel by allowing relaxation along the unguided y coordinate.

As shown in Fig. 5.2, PathGennie successfully generated transitions through both the upper and lower saddle channels. The distribution of paths was sensitive to the runner length: short τ_2 values produced trajectories that crossed through both channels with roughly equal probability, whereas longer τ_2 values allowed the system to relax along the unguided y degree of freedom, shifting the distribution toward the lower-energy channel. This result demonstrates that PathGennie can discover transition pathways even when guided by an incomplete progress coordinate, provided the runner segment is sufficiently long to permit structural relaxation in orthogonal degrees of freedom.

5.6 Ligand Unbinding Pathways

Ligand dissociation from proteins is a paradigmatic rare event that is both scientifically important and methodologically challenging. Unbinding rates span many orders of magnitude, and the underlying pathways often involve complex rearrangements of both ligand and receptor. We applied PathGennie to two benchmark systems: benzene unbinding from T4 lysozyme L99A, a well-characterised model for hydrophobic cavity binding, and imatinib dissociation from Abl kinase, a clinically relevant system featuring slow, multi-route unbinding.

5.6.1 Benzene Dissociation from T4 Lysozyme L99A

The L99A mutant of T4 lysozyme contains a buried hydrophobic cavity that binds small aromatic ligands, making it a classic model system for studying ligand–protein interactions and unbinding kinetics [**Eriksson1992**, **Morton1995**, **Baase2010**]. Experimental measurements report dissociation rates on the millisecond timescale and previous simulation studies suggested multiple routes for benzene egress involving different helical elements [**Buch2011**, **Niitsu2019**, **Vallurupalli2016**, **NunesAlves2018**, **NunesAlves2021**, **Wang2016**, **Rydzewski2019**, **Ray2024**].

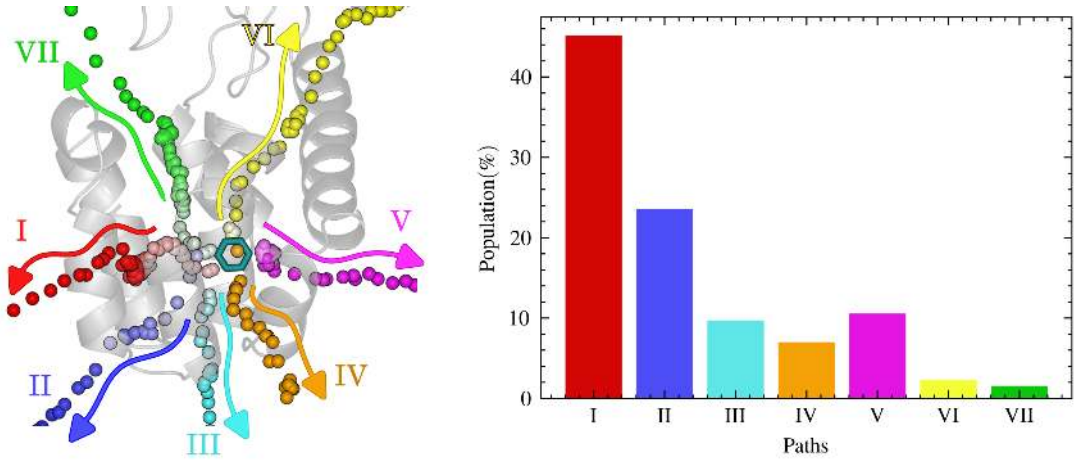

Figure 5.3: Benzene unbinding pathways from T4 lysozyme L99A. Representative trajectories from the seven identified exit route clusters are shown around the binding cavity, together with the relative population of each cluster.

Using the PCA-based distance coordinates in escape mode, we generated 1000 unbinding trajectories with $N_{\text{trial}} = 25$, $\tau_1 = 0.01$ ps, and $\tau_2 = 0.05$ ps. The average reactive length was 98±22 ps, with a total computational cost of approximately 228±56 ps per trajectory (Table 5.1). Clustering the trajectories at the point of last protein contact resolved seven distinct exit pathways (Fig. 5.3). The dominant route, accounting for 68% of trajectories, passed between helices F, G, and H, consistent with prior biased MD and random acceleration molecular dynamics simulations [**NunesAlves2021**, **Wang2016**].

The remaining trajectories resolved secondary exit channels, including routes between helices E and F and over the C-terminus of helix G, demonstrating that PathGennie can resolve pathway heterogeneity even in a relatively simple binding site.

5.6.2 Imatinib Unbinding from Abl Kinase

Imatinib binds the inactive DFG-out conformation of Abl kinase, stabilising an activation-loop-folded state that is distinct from the active conformation [**Ayaz2023**]. Dissociation is slow (tens of milliseconds), and prior computational studies have identified exit pathways passing near the ATP-binding cleft, the P-loop, and the αC-helix [**Narayan2021**, **Paul2020**, **Shekhar2022**, **Yang2009**, **Oruganti2021**]. The existence of multiple competing routes, coupled with large-scale conformational changes in the kinase domain, makes this system an ideal test for PathGennie's ability to discover complex, coupled ligand–receptor exit mechanisms.

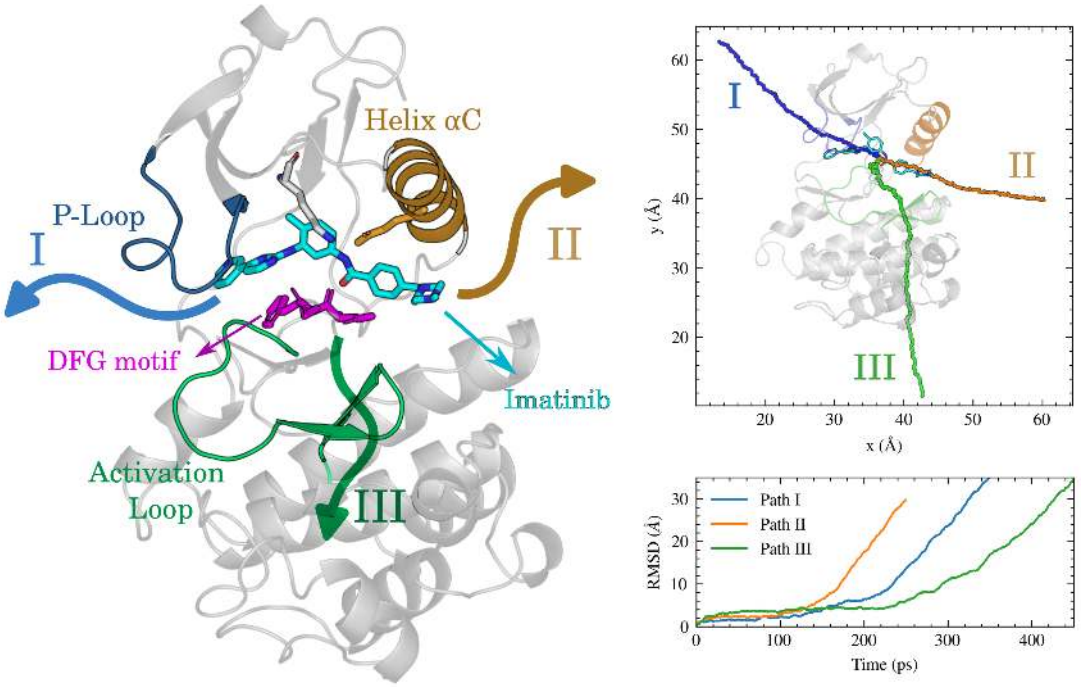

Figure 5.4: Imatinib dissociation from Abl kinase. PathGennie identifies three main exit channels relative to the activation loop, P-loop, and αC-helix. Ligand RMSD traces (right) show the corresponding dissociation progress for representative trajectories from each channel.

PathGennie was applied using the 5D PCA distance coordinate with $N_{\text{trial}} = 25$, $\tau_1 = 0.02$ ps, and $\tau_2 = 0.03$ ps. The algorithm recovered three exit channels (Fig. 5.4) with an average reactive length of 184 ± 49 ps. Analysis of 500 trajectories revealed that the ATP-cleft route dominated (57%), followed by the P-loop route (41%) and the αC-helix route (2%). Notably, the unbinding trajectories captured large-scale conformational changes in the P-loop and αC-helix that were not explicitly encoded in the CV space. This

emergent coupling between ligand motion and receptor rearrangement demonstrates that unbiased trial segments can reveal mechanistic details that might be missed by strongly biased methods that constrain the receptor.

5.7 Folding and Unfolding of Fast-Folding Proteins

To test PathGennie on systems with well-defined native and denatured basins, we applied the directed mode to the folding and unfolding of two fast-folding proteins: the 20-residue Trp-cage miniprotein and the 56-residue B1 domain of streptococcal protein G. For these systems, the progress coordinate combined two physically motivated order parameters: the fraction of native contacts Q and the radius of gyration R_g. The native-contact fraction was computed using a continuous switching function:

$$Q(X) = \frac{1}{|S|} \sum_{(i,j)\in S} \frac{1}{1 + \exp[\beta(r_{ij}(X) - \lambda r_{ij}^0)]}, \tag{5.3}$$

where S is the set of native heavy-atom pairs separated by more than three residues and closer than 4.5 Å in the native structure, with parameters $\beta = 5$ Å^{-1} and $\lambda = 1.8$ **[Best2013]**. For folding simulations, we additionally included the C_α RMSD to the native state to prevent collapse into non-native compact states.

For both proteins, unfolding transitions were generated on sub-nanosecond reactive timescales (150 ± 33 ps for Trp-cage, 176 ± 67 ps for Protein G), representing acceleration factors of 10^3–10^4 relative to experimental unfolding timescales. Folding transitions required longer trajectories (336 ± 95 ps for Trp-cage, 612 ± 121 ps for Protein G) but were still orders of magnitude shorter than the corresponding experimental folding times. Figure 5.5 shows these generated folding and unfolding paths on reference free-energy surfaces computed from long unbiased trajectories **[LindorffLarsen2011, Piana2011]**. In that projection, the PathGennie trajectories closely followed the minimum-free-energy folding pathways, suggesting that the algorithm does not generate artificial high-energy shortcuts but rather captures the intrinsic folding mechanism.

5.8 The Path-Resolved Kinetics Workflow

Although PathGennie can readily generate candidate pathways, it does not directly yield quantitative kinetic rates. The path-resolved kinetics workflow transforms the raw path ensemble into rigorous, route-specific rate constants through five sequential stages, as shown in Fig. 5.6:

1. Generate a diverse path ensemble using PathGennie. Use Dynamic Time Warping (DTW) hierarchical clustering to group paths into mechanistically distinct families.

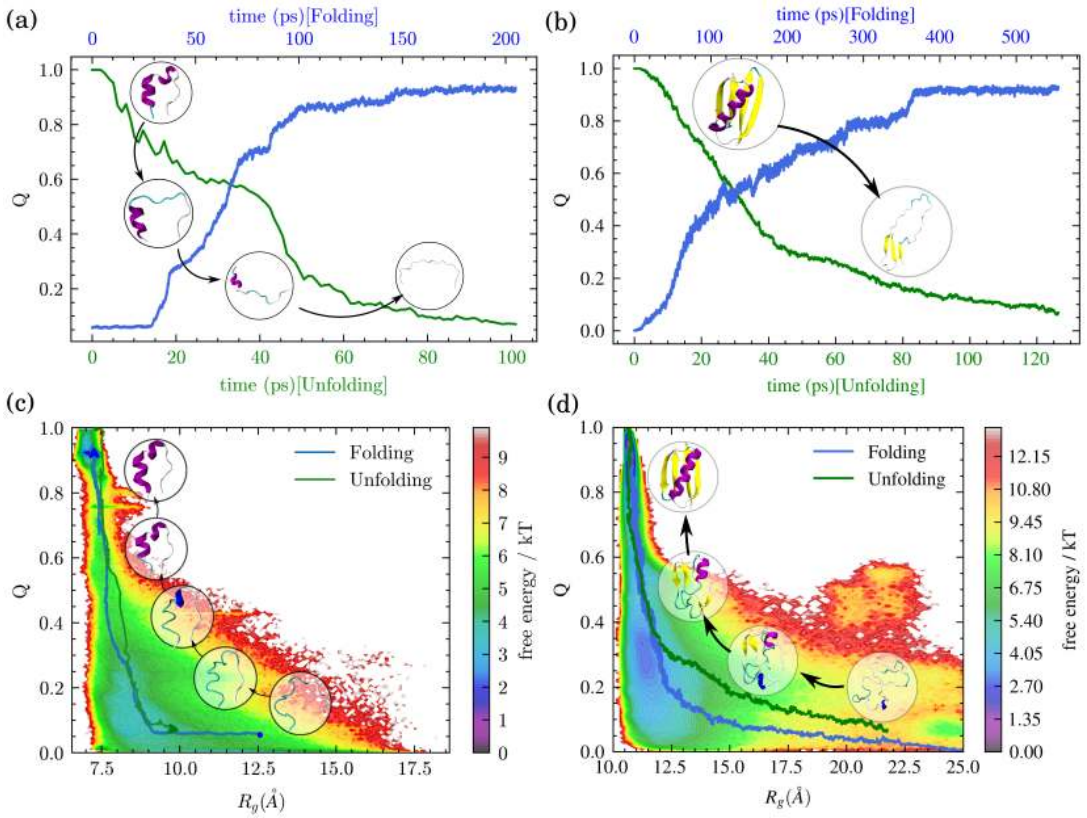

Figure 5.5: Folding and unfolding paths for Trp-cage and Protein G. Generated trajectories show the expected change in native-contact content, and when projected onto reference free-energy surfaces constructed from long unbiased simulations, they follow the minimum-free-energy pathways.

item Train a neural network to refine each family into a smooth representative reference path.

2. Construct a path collective variable (PathCV) for each refined reference path.
3. Run independent, pre-seeded WE simulations along each PathCV to estimate route-specific rate constants.

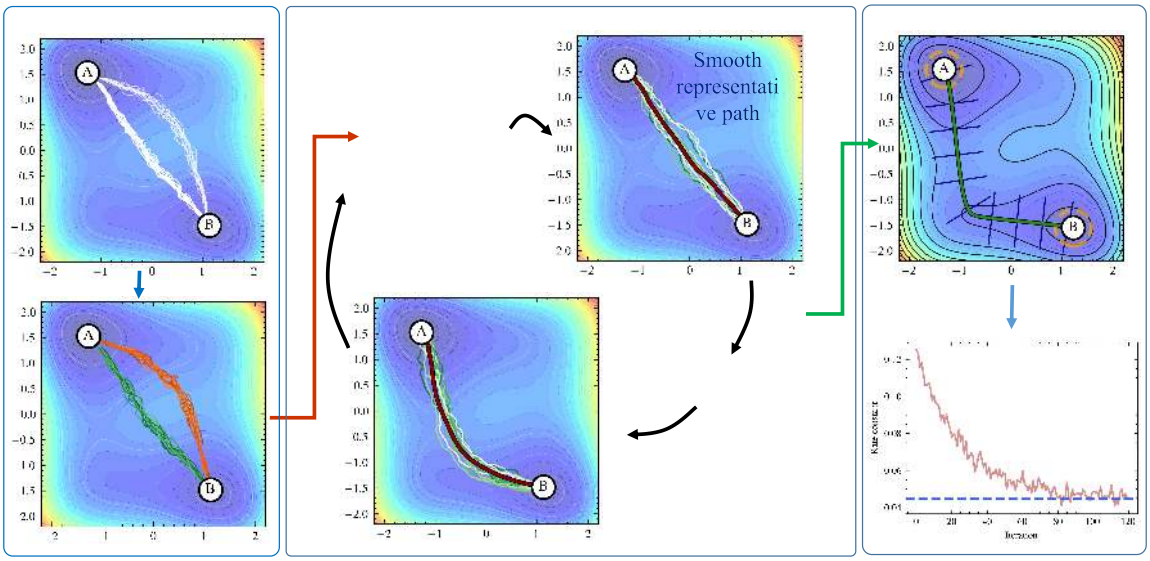

Figure 5.6: Overview of the path-resolved weighted-ensemble workflow. (a) Initial reactive trajectories are generated with PathGennie. (b) Trajectories are clustered into mechanistically distinct path families using DTW. (c) For each family, a neural-network representation smooths and reparameterises the reference path. (d) The refined reference path defines PathCVs along which PathGennie is rerun until stabilisation. (e) The converged PathCV is used to discretise configuration space for pre-seeded WE simulations, yielding (f) pathway-specific kinetic observables.

5.8.1 Dynamic Time Warping for Path Classification

Because transition paths can vary in both duration and the sequence of conformational changes, standard Euclidean alignment in time is inappropriate for comparing trajectories. DTW provides a flexible alignment framework that accounts for temporal elasticity by allowing non-linear mappings between time indices, enabling meaningful comparison of pathways that proceed at different rates but follow similar configurational routes [**Sakoe1978**, **Elangovan2025**, **Ray2024**]. This makes DTW particularly well suited for analysing transition-path ensembles from rare-event simulations. For two trajectories $A = \{a_1, \ldots, a_m\}$ and $B = \{b_1, \ldots, b_n\}$ in CV space, DTW finds the alignment path π that minimises the cumulative distance:

$$D_{\mathrm{DTW}}(A, B) = \min_{\pi} \sum_{(p,q)\in\pi} d(a_p, b_q), \tag{5.4}$$

where $d(a_p, b_q)$ is the Euclidean distance in CV space. We employed a Sakoe-Chiba band constraint restricting the alignment to within 20% of the longer trajectory length, preventing pathological alignments that match distant time points [**Sakoe1978**]. The resulting DTW distance matrix was assembled and subjected to average-linkage hierarchical agglomerative clustering [**scikit_learn**], yielding a dendrogram from which distinct pathway families were identified. Sparsely populated families (such as the fourth trypsin route) were excluded from PathCV construction because the available trajectory data did not provide sufficient training material for a physically transferable path representation.

5.8.2 Neural Network Path Refinement and PathCV Construction

Raw MD trajectories are noisy and unevenly sampled, and direct use of such trajectories would produce poorly conditioned path collective variables. For each trajectory cluster, a continuous reference path is therefore first constructed through neural network fitting. All trajectories assigned to a cluster are reparameterised onto a normalised progress coordinate $t \in [0, 1]$ and aggregated into pairs $(t, \mathbf{x})$, where $\mathbf{x}$ is the molecular configuration or its projection in the chosen CV space. A fully connected feed-forward neural network is trained to learn the mapping from t to $\mathbf{x}$, minimising mean squared error (MSE) with the Adam optimiser (lr = 10^{-3}, 500 epochs, SiLU activations in hidden layers, linear output layer). The default architecture is 1–128–128–d for low-dimensional model systems, and 1–512–512–d or larger for biomolecular PCA spaces, where d is the CV dimensionality. The trained network is sampled at uniformly spaced values of t, and each sampled point is mapped to the nearest realised configuration from the trajectory ensemble, yielding an evenly parameterised path that suppresses thermal noise while preserving the essential geometric progression.

For each refined reference path, the PathCV is constructed following the formulation of Branduardi et al. [**Branduardi2007**]. Given N path nodes $\{X_i\}_{i=1}^{N}$, let D_i denote the mean-squared displacement from the instantaneous configuration X to node X_i in the chosen CV space. The unnormalised progress coordinate s_{raw}, normalised progress $s \in [0, 1]$, and orthogonal deviation z are:

$$s_{\mathrm{raw}} = \frac{\sum_{i=1}^{N} i \exp(-\lambda D_i)}{\sum_{i=1}^{N} \exp(-\lambda D_i)}, \tag{5.5}$$

$$s = \frac{s_{\mathrm{raw}} - 1}{N - 1}, \tag{5.6}$$

$$z = -\frac{1}{\lambda} \ln\left[\sum_{i=1}^{N} \exp(-\lambda D_i)\right]. \tag{5.7}$$

The normalisation in Eq. (5.6) maps the first and final path nodes to $s = 0$ and $s = 1$,

respectively. For approximately equidistant path nodes, the smoothing parameter λ is set as:

$$\lambda = \frac{2.3}{\langle D(X_{i+1}, X_i)\rangle_{i=1}^{N-1}}, \tag{5.8}$$

where $D(X_{i+1}, X_i)$ is the mean-squared displacement between adjacent path nodes, and λ is tuned manually only when required to maintain overlap between neighbouring nodes. The coordinate s tracks how far a configuration has advanced along the reference path, while z monitors perpendicular deviation, providing a natural coordinate for WE binning that confines walkers to the transition tube surrounding the reference curve.

5.8.3 Iterative Path Refinement and Convergence

If a suboptimal progress coordinate is used to generate the first reactive trajectories, the resulting reference path may be displaced from the physically relevant transition channel. The path-generation step itself can then function as a refinement operator: an initial trajectory ensemble is clustered and smoothed as above, the resulting PathCV is used to generate a new ensemble around the reference path, and the procedure is iterated until consecutive reference paths stabilise. This iterative refinement is monitored using the discrete Fréchet distance [**Efrat2002**] between consecutive reference paths, which compares ordered curves and decreases monotonically until it plateaus at convergence.

Figure 5.7(a,b) illustrates the refinement on the Müller–Brown potential. Starting from a coarse initial path, alternating cycles of path generation and neural network refitting drove the reference coordinates toward the minimum-free-energy isocommittor pathway (zero-temperature-string reference [**E2002**, **E2005**, **E2007**]), and the Fréchet distance between consecutive paths decreased monotonically until reaching a stable plateau.

We next applied the same iterative protocol to the OAMe-G2 host–guest system, which contains two mechanistically distinct unbinding routes: a *dry pathway* in which the guest escapes without water penetration into the host cavity, and a *wet pathway* in which a water molecule participates directly in the dissociation event. A simple host–guest centre-of-mass distance generated predominantly dry-like trajectories, whereas the PCA-based CV produced a richer ensemble containing both wet and dry events (Fig. 5.7(d)). The wet subset was iteratively refined; convergence was obtained after seven iterations (Fig. 5.7(e)). The resulting PathCV was then used for short well-tempered metadynamics (WT-MetaD) simulations. The reweighted free-energy surface obtained from WT-MetaD is consistent with the brute-force zero-temperature-string reference for the wet channel, confirming the physical relevance of the refined path (Fig. 5.7(f)). The brute-force/ZTS reference was obtained from a zero-temperature string calculation performed on a free-energy surface reconstructed from a 500 ns OPES simulation biasing a Deep-LDA CV [**Rizzi2021**]. The OAMe-G2 result illustrates the power of iterative refinement:

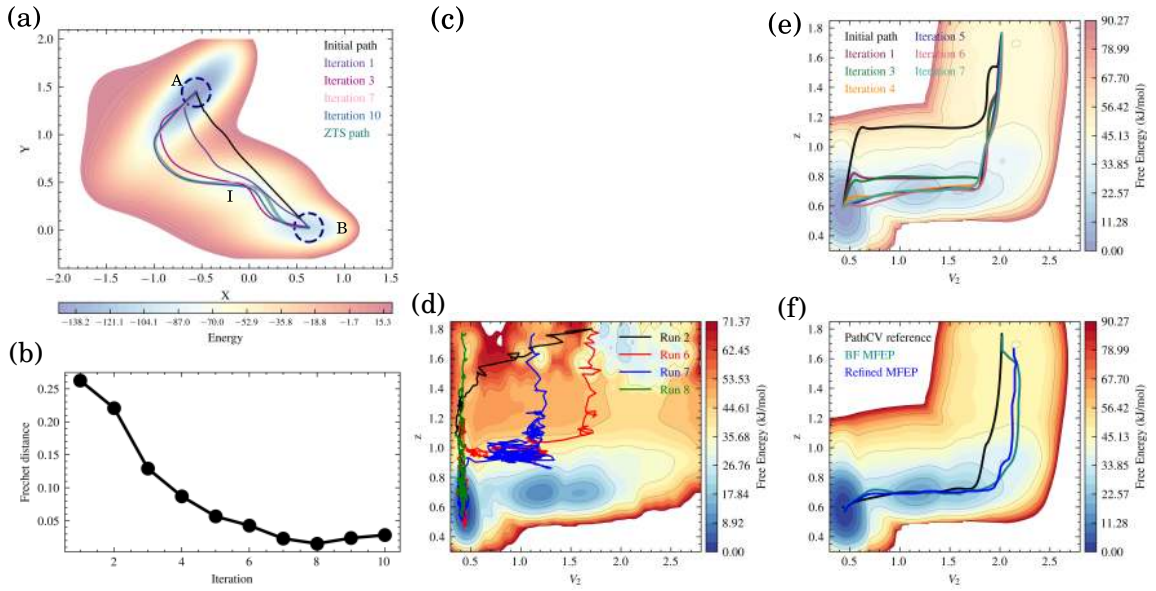

Figure 5.7: Iterative refinement of reactive PathGennie paths. (a) Successive paths on the Müller–Brown potential approach the zero-temperature-string reference path, with source (A), target (B), and intermediate (I) annotated. (b) Discrete Fréchet distance between consecutive paths, monitoring convergence. (c) Representative view of the synthetic OAMe–G2 host–guest system; guest G2 is shown in cyan sticks. (d) PathGennie trajectories in PCA space projected onto the physical z–V_2 plane, where z is the ligand center-of-mass projection on the binding axis and V_2 is a water-coordination order parameter distinguishing wet and dry channels. (e) Iterative refinement of the wet OAMe–G2 dissociation path over seven iterations. (f) Reweighted free-energy surface from WT-MetaD using the refined wet-path PathCV, compared with the brute-force/ZTS reference pathway.

starting from a simple geometric CV that cannot distinguish wet from dry dissociation, the framework converges onto the physically meaningful wet channel and validates it against an independent reference.

Several established methods including nudged elastic band, finite-temperature string, and swarm-of-trajectories approaches can refine trial pathways toward minimum-free-energy paths [**Henkelman2000**, **E2002**, **E2007**, **Pan2008**]. The present use is complementary but narrower: the same path-generation framework that produces the initial ensemble serves as a practical refinement operator, so that pathway identification, PathCV construction, and WE initialisation remain internally consistent.

5.8.4 Pre-Seeded WE Sampling Along Path CVs

The weighted ensemble method propagates many weighted trajectory replicas in parallel and periodically redistributes statistical effort through splitting and merging operations, thereby efficiently sampling low-probability transition regions while preserving unbiased kinetics [**Huber1996**, **Zuckerman2017**]. Under steady-state conditions, walkers that reach the target state are recycled back into the source basin while retaining their

statistical weights [**Bhatt2010**], and the steady-state flux into the target state directly yields the transition rate constant. In the present work, WE sampling is integrated with PathCVs constructed from the refined transition pathways. The converged reference path is discretised into a sequence of reference frames that define the progress coordinate s, these serve as mechanistically informed initialisation points, and the associated s–z binning structure confines walkers to the relevant transition tube while permitting orthogonal fluctuations.

Unlike conventional WE initialisation that populates only the source basin, pre-seeding walkers directly along the refined path ensemble ensures immediate representation of both metastable basins and intermediate transition regions. Each walker is initially assigned an equal statistical weight ($1/N$), after which standard WE splitting and merging redistribute probability toward the correct steady-state distribution. For pathway p, the steady-state rate constant is estimated from the long-time-average recycled flux:

$$k_p = \lim_{N\to\infty} \frac{1}{N\tau} \sum_{i=1}^{N} F_p^{(i)}, \tag{5.9}$$

where $F_p^{(i)}$ is the total statistical weight reaching the target state during WE iteration i and τ is the resampling interval. When pathways are analysed as competing channels sharing the same source and target states, the total rate is approximated as $k_{\text{tot}} \approx \sum_p k_p$ for non-overlapping channels with a common source normalisation.

A critical subtlety concerns *channel switching* and the relation between this decomposition and transition-path theory (TPT). TPT decomposes the equilibrium reactive current using forward and backward committors and can describe recrossing throughout configuration space [**Metzner2009**, **E2010**]. The present decomposition is imposed by trajectory-derived spatial tubes and measured only through steady-state target flux; the two coincide when the tubes form non-overlapping, effectively no-recrossing partitions of the reactive ensemble. When tubes overlap, walkers can switch channels and lose their channel identity through merging [**Zhang2010**, **Bose2025**]. We mitigate this in the trypsin system by binning in both s and z and by running channels separately; this is a system-specific mitigation rather than a general correction. Strongly overlapping pathways require history-labelled bins or a joint multi-channel WE calculation.

WE parameters were chosen to balance resolution along the PathCV with statistical efficiency. For the model three-hole potential: $N = 45$ bins, $M = 4$ walkers per bin, $\tau = 50$–100 integration steps. For the 1OPJ Abl-imatinib system: $N = 45$, $M = 3$, $\tau = 40$ ps. For the 3PTB benzamidine-trypsin system, closely spaced pathways necessitated additional discretisation along z: $N = 25$, $M = 3$, $\tau = 20$ ps, plus three exponentially-spaced bins along z.

Uncertainty was estimated from three independent WE replicates using a Bayesian

bootstrap: replicate rate values were reweighted with Dirichlet-distributed weights, producing bootstrap samples of the mean rate, and the 95% confidence interval (CI) was taken as the 2.5th–97.5th percentile interval averaged over the converged analysis window. Because only three independent replicates were available, these intervals should be interpreted as replicate-level uncertainty estimates rather than as fully converged sampling-error bounds.

5.9 Path-Informed Weighted Ensemble Simulations

Before applying the full path-resolved kinetics workflow to complex multi-pathway systems, we first validated the core idea of path-informed WE initialisation on well-characterised single-channel benchmarks: the $C_{7\mathrm{eq}} \rightarrow \alpha_L$ transition of alanine dipeptide in explicit solvent, and the folding of the CLN025 chignolin variant in implicit solvent. For alanine dipeptide, we employed a 36×36 grid in the ϕ–ψ plane with a 2 ps resampling interval. For chignolin, we used one-dimensional bins along C_α RMSD with a 20 ps resampling interval [**Bogetti2019**]. In the path-informed protocol, walkers were initialised by sampling structures uniformly along PathGennie-generated transition paths, with each walker assigned a uniform initial weight.

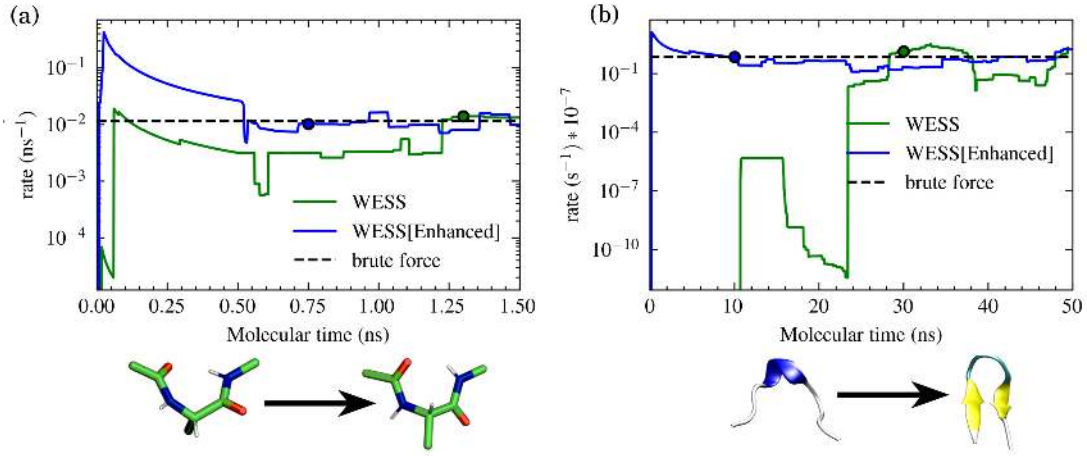

Figure 5.8: Rate convergence in path-informed WE versus standard source-only WE. Pre-seeding walkers along PathGennie-generated pathways accelerates the attainment of stable rate estimates for (a) alanine dipeptide and (b) chignolin.

Path-informed WE achieved stable rate constants substantially faster than source-only WE (Fig. 5.8). For alanine dipeptide, convergence was reached after 612 ns of aggregate simulation, compared to 1.7 μs for standard WE; this corresponds to a 2.8-fold reduction in total cost. For chignolin, the path-informed run converged in 350 ns, whereas standard WE required 1.4 μs, yielding a 4-fold speedup. These gains arise because pre-seeding walkers across the transition region bypasses the transient discovery phase, allowing the WE algorithm to immediately focus computational effort

on the rare barrier-crossing ensemble.

5.10 Applications of Path-Resolved WE

We applied the complete path-resolved workflow to three systems of increasing complexity: a model potential with temperature-dependent pathway crossover, a clinically relevant kinase mutation that alters unbinding routes, and a standard benchmark for ligand dissociation kinetics. These applications demonstrate that the framework can resolve mechanistic questions that are inaccessible to standard single-coordinate methods.

5.10.1 Temperature Crossover in a Three-Hole Potential

As a first validation, we examined the two-dimensional three-hole potential:

$$U(x,y) = -3.0\,e^{-x^2}\left[e^{-(y-5/3)^2} - e^{-(y-1/3)^2}\right] - 5.0\,e^{-y^2}\left[e^{-(x-1)^2} + e^{-(x+1)^2}\right], \tag{5.10}$$

which features two deep minima near $(-1, 0)$ and $(1, 0)$ serving as reactant and product states (A and B), and a shallower intermediate basin C above them. This topology gives rise to two competing transition routes: an indirect upper channel passing through the intermediate (ACB route) and a direct lower channel (AB route). The dominant mechanism is known to depend on temperature: at low temperature, the reaction is controlled primarily by energetic considerations and proceeds preferentially through the upper route despite its longer geometric length, because it traverses lower-energy saddle regions; at higher temperatures, the entropic and diffusive contributions favour the geometrically shorter lower channel [**Park2003**, **Huo1997**].

PathGennie-generated reactive trajectories were clustered into two families corresponding to the upper and lower channels, and independent WE simulations were performed using the PathCV associated with each. At $T = 50$ K, the upper (ACB) pathway carried the majority of the reactive flux and converged to a substantially larger rate than the lower (AB) channel (Fig. 5.9(c)). Increasing the temperature to $T = 200$ K produced a clear reversal: the lower channel became increasingly accessible and ultimately carried the dominant reactive flux (Fig. 5.9(d)).

A useful comparison is provided by conventional WE simulations performed using only the x coordinate of the potential (Fig. 5.9(e,f)). Because this projection cannot distinguish between the upper and lower routes, the resulting kinetics represent a mixture of both channels, and at low temperatures the rate failed to converge. Using a two-dimensional (x, y) progress coordinate, the overall transition rate can be estimated, but no mechanistic information regarding pathway utilisation is retained. In contrast, the PathCV-based decomposition resolves the individual flux contributions and directly

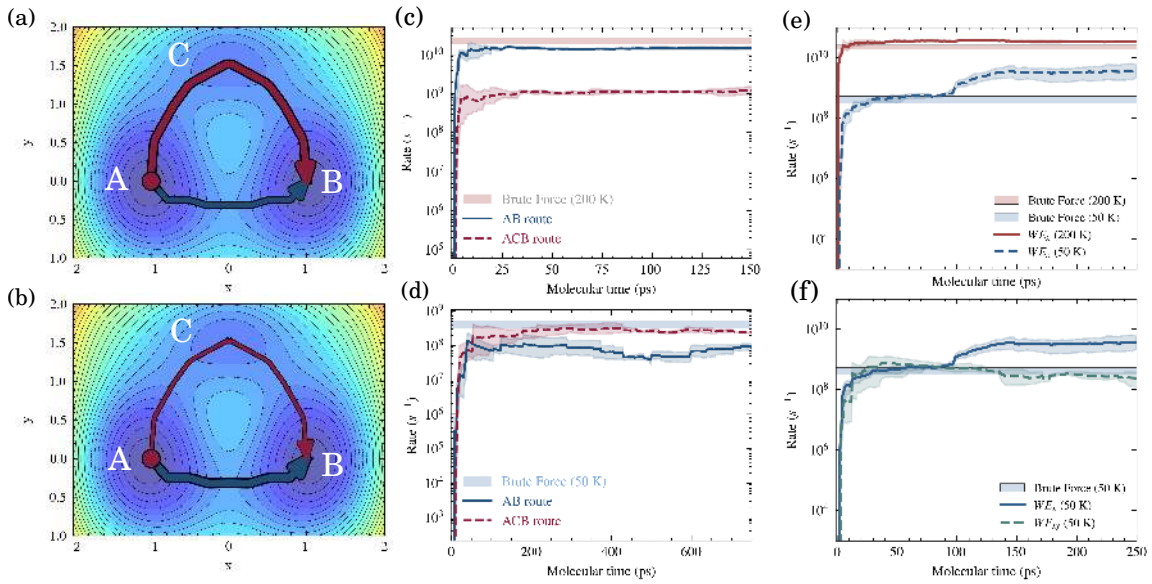

Figure 5.9: Temperature-dependent pathway crossover in the three-hole potential. (a, b) Representative transition pathways at $T = 50$ K and $T = 200$ K. (c, d) Convergence of pathway-specific transition rates through the upper and lower channels at each temperature. (e) Rate estimates from WE using a suboptimal one-dimensional x coordinate, which fails to converge at low temperature. (f) Low-temperature rate convergence comparing the two-dimensional (x, y) representation with the one-dimensional coordinate.

reveals the temperature-induced redistribution of reactive current between competing channels, a mechanistic crossover that single-coordinate WE cannot detect. This three-hole system therefore serves as a controlled proof-of-principle demonstrating that pathway-specific WE simulations can capture phenomena that are hidden in conventional single-coordinate kinetic analyses.

5.10.2 Mutation-Dependent Imatinib Dissociation from Abl Kinase

The therapeutic targeting of kinases has been a central strategy in modern drug discovery, with Abelson tyrosine kinase (Abl) serving as a prototypical system [**Roskoski2015**, **Levitzki2003**, **Sridhar2000**]. Imatinib (Gleevec) is the first generation Abl inhibitor with great clinical success, but drug resistance remains a major issue [**Druker2001**, **Druker2006**, **Chandrasekhar2019**, **Hoemberger2020**, **Soverini2014**]. Resistance is often caused by single point mutations in the kinase domain resulting in altered conformational dynamics and not necessarily altered binding affinity [**Nagar2003**].

Work by Lyczek et al. showed a kinetic mechanism of resistance in the N368S mutant, where imatinib has a similar binding affinity (K_d) to the wild-type (WT) protein, but dissociates roughly three times faster [**Lyczek2021**]. This behaviour has been reproduced computationally using infrequent metadynamics, with kinetic estimates in agreement with experiment [**Shekhar2022**, **Lee2024**]. These results highlight the significance of ligand residence time as a determinant of drug efficacy: resistance can

be driven by kinetic factors rather than thermodynamic stability alone.

Imatinib dissociation from the Abl ATP-binding site proceeds through multiple pathways. Computational studies have identified two dominant exit routes: (i) a pathway passing beneath the P-loop toward the αC helix (here referred to as the αC pathway), and (ii) a pathway closer to the kinase hinge region (the P-loop/hinge pathway). It has been suggested that pathway preference can be altered upon mutation, with N368S enhancing access to the αC-associated channel due to altered local interactions and increased conformational flexibility [**Shekhar2022**].

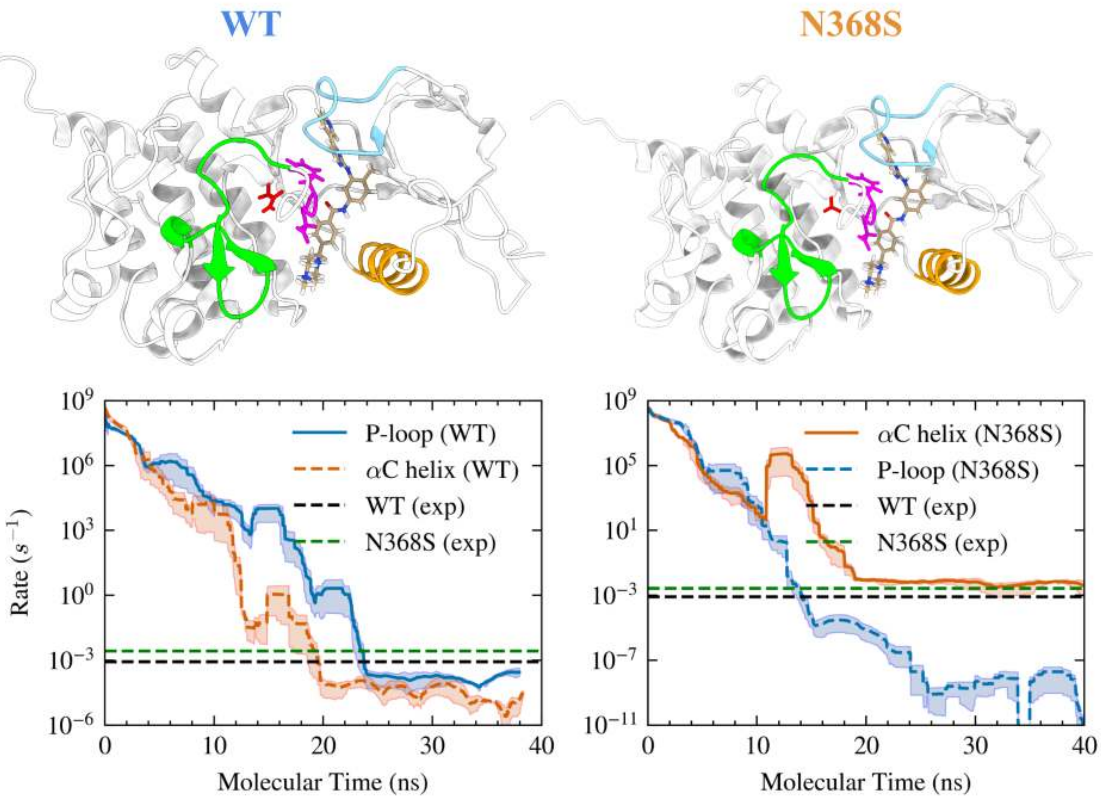

Figure 5.10: Path-resolved imatinib dissociation from WT and N368S Abl kinase. Structural representations of the imatinib-bound complex for (a) WT and (b) N368S, with the mutation site in red, DFG motif in magenta, αC helix in orange, P-loop in sky blue, and activation loop in lime green. Panels (c) and (d) show the pathway-specific dissociation rates for WT and N368S, respectively, as functions of molecular time. In WT, the P-loop/hinge route carries the dominant kinetic flux; mutation redistributes this flux overwhelmingly to the αC-helix channel.

We directly probed this mutation-dependent pathway crossover by computing pathway-specific dissociation rates using the path-resolved WE framework. Figure 5.10 summarizes the structural pathways and their time-dependent rate estimates, while Table 5.2 reports the converged pathway-specific rates. DTW clustering of PathGennie trajectories from PDB 1OPJ identified two primary exit routes. For WT Abl, path-resolved WE simulations over a converged 34–36 ns molecular-time window revealed that the

P-loop/hinge route carries the larger kinetic flux, with $k_{\text{off}}^{\text{P-loop}} = 1.101 \times 10^{-4}$ s^{-1} (95% CI: $[8.804 \times 10^{-5}, 1.353 \times 10^{-4}]$ s^{-1}), while the αC route is slower at $k_{\text{off}}^{\alpha\text{C}} = 1.269 \times 10^{-5}$ s^{-1} (95% CI: $[9.478 \times 10^{-6}, 1.675 \times 10^{-5}]$ s^{-1}).

In the N368S mutant, the flux redistribution is dramatic (converged window 38–40 ns): the P-loop/hinge pathway is effectively suppressed ($k_{\text{off}}^{\text{P-loop}} = 6.201 \times 10^{-9}$ s^{-1}; 95% CI: $[3.010 \times 10^{-10}, 1.215 \times 10^{-8}]$ s^{-1}), while the αC route becomes the dominant channel with $k_{\text{off}}^{\alpha\text{C}} = 5.774 \times 10^{-3}$ s^{-1} (95% CI: $[2.896 \times 10^{-3}, 8.533 \times 10^{-3}]$ s^{-1}). The WT P-loop rate lies within roughly an order of magnitude of both the infrequent-metadynamics estimate [**Shekhar2022**] and experiment [**Lyczek2021**]; the mutation redirects the dominant reactive channel from the P-loop/hinge route to the faster αC route. We emphasise the reproducible mechanistic crossover rather than the precise experimental rate ratio, which the present replicate statistics do not yet fully constrain.

System	Pathway	Infreq. Metad. k_{off} [s^{-1}]	Experimental k_{off} [s^{-1}]	WE-PathCV k_{off} [s^{-1}]	Bootstrap 95% CI [s^{-1}]
Wild-type	P-loop	$(6 \pm 3) \times 10^{-4}$	$(8.3 \pm 0.83) \times 10^{-4}$	1.101×10^{-4}	$[8.804 \times 10^{-5}, 1.353 \times 10^{-4}]$
	αC helix			1.269×10^{-5}	$[9.478 \times 10^{-6}, 1.675 \times 10^{-5}]$
N368S	P-loop	$(4 \pm 2) \times 10^{-3}$	$(2.7 \pm 0.27) \times 10^{-3}$	6.201×10^{-9}	$[3.010 \times 10^{-10}, 1.215 \times 10^{-8}]$
	αC helix			5.774×10^{-3}	$[2.896 \times 10^{-3}, 8.533 \times 10^{-3}]$

Table 5.2: Path-specific imatinib dissociation rates from WT and N368S Abl kinase estimated with the WE-PathCV workflow, compared with infrequent metadynamics [**Shekhar2022**] and experiment [**Lyczek2021**]. WE-PathCV uncertainties are Bayesian-bootstrap 95% confidence intervals over three independent replicates in the converged analysis window. Aggregate WE simulation cost per path: ≈ 5.3 μs.

A crucial conceptual point concerns the distinction between *geometric pathway population* and *kinetic flux*. In the PathGennie trajectory ensembles, the αC channel is the more frequently generated route in both WT and N368S systems; yet in WT, it is the less-populated P-loop/hinge channel that carries the dominant kinetic flux. A highly populated channel can therefore be kinetically subdominant: geometric population during path sampling reflects the ease of generating trial configurations along that route, whereas WE flux measures the rate at which reactive trajectories actually traverse the channel and escape. The two measures move in the same qualitative direction upon mutation: the hinge channel becomes both less populated and far less reactive in N368S, but only the path-resolved WE rates quantify the redistribution of reactive current that underlies the experimentally observed acceleration of imatinib dissociation. This behaviour can be rationalised through mutation-induced changes in the local interaction network around the DFG motif: the N368S substitution weakens stabilising interactions along the αC exit route, effectively lowering the kinetic bottleneck for that channel [**Lyczek2021**, **Shekhar2022**].

5.10.3 Path-Resolved Dissociation Kinetics of Benzamidine from Trypsin

Benzamidine dissociation from trypsin is a canonical benchmark system for rare-event simulation methods, with an experimentally measured unbinding rate of 600 ± 300 s^{-1} [**Guillain1970**, **Marquart1983**, **Schiebel2018**]. The rate-limiting process is ligand dissociation on the millisecond timescale. A wide range of computational approaches have been applied to this system, including Markov state models, metadynamics, Brownian dynamics, REVO, WExplore, and Markovian Weighted Ensemble Milestoning [**Teo2016**, **Tiwary2015**, **Buch2011**, **Plattner2015**, **Votapka2017**, **Betz2019**, **Doerr2014**, **Brotzakis2018**, **Ansari2022**, **Ray2021**, **Donyapour2019**, **Dickson2014**, **Dickson2017**, **Sohraby2023**, **Kokh2018**, **Ding2026**]. Previous studies have demonstrated that benzamidine exits through multiple distinct pathways, but the relative flux through each route has remained unresolved.

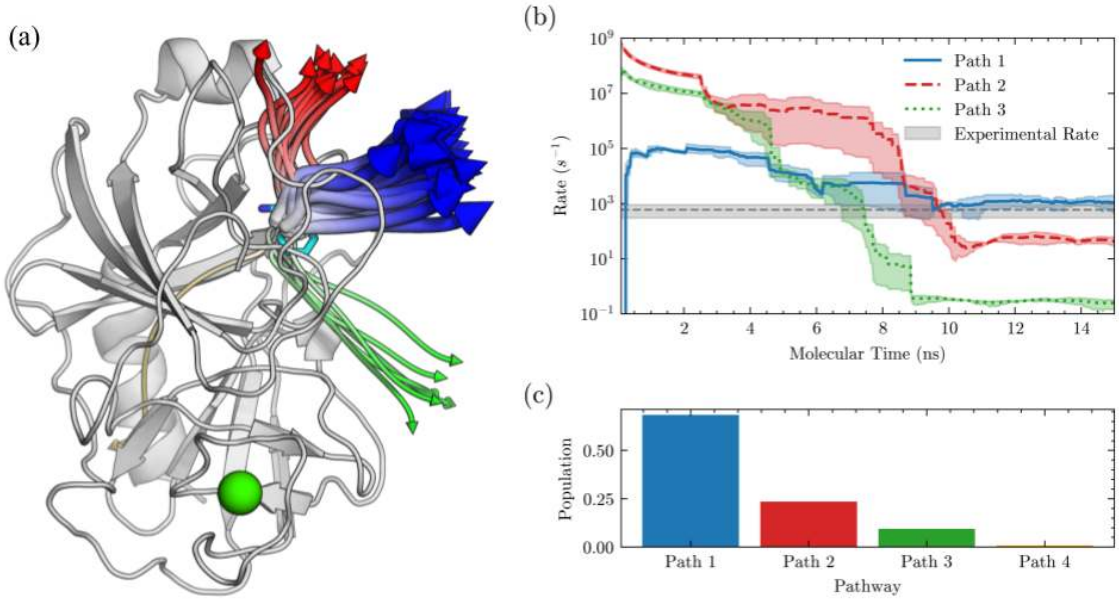

Figure 5.11: Path-resolved dissociation kinetics of benzamidine from trypsin (PDB 3PTB). (a) Major unbinding pathways identified from trajectory clustering shown as coloured tubes; tube radius encodes PathGennie trajectory population (not kinetic flux). The protein is shown in grey cartoon, with calcium as a green sphere. (b) Time evolution of pathway-specific dissociation rates from WE simulations. (c) Relative pathway populations in the PathGennie ensemble (colour consistent with panel (a)). Path 1 dominates the reactive flux by more than an order of magnitude over Path 2, and Path 3 is negligible.

PathGennie was applied to PDB 3PTB using τ_1 = 40 fs, τ_2 = 100 fs, and N_{trial} = 50, generating a large ensemble of reactive trajectories. DTW clustering identified three dominant dissociation pathways and a fourth, sparsely populated route (Fig. 5.11(c));

the fourth pathway was excluded from PathCV construction. Several of the identified pathways overlap with those previously reported using WExplore and REVO [**Dickson2017**, **Donyapour2019**]. After constructing PathCVs for the three major pathways, independent WE simulations ($N = 25$ bins, $M = 3$, $\tau = 20$ ps, plus three exponentially-spaced z-bins) were performed over a converged 14–15 ns molecular-time window. The resulting pathway-specific rate constants are given in Table 5.3.

Path	Rate [s^{-1}]	Bootstrap 95% CI [s^{-1}]	Replicate rates [s^{-1}]
Path 1 (dominant)	1.153×10^3	$[7.222 \times 10^2, 1.791 \times 10^3]$	735.9, 741.6, 1985.5
Path 2 (secondary)	4.649×10^1	$[3.466 \times 10^1, 5.883 \times 10^1]$	45.8, 31.5, 62.2
Path 3 (negligible)	2.572×10^{-1}	$[1.537 \times 10^{-1}, 3.370 \times 10^{-1}]$	0.356, 0.292, 0.123

Table 5.3: Path-specific benzamidine–trypsin unbinding rates from WE-PathCV simulations, evaluated over the converged 14–15 ns molecular-time window. Bayesian-bootstrap 95% CI from three independent WE replicates. Aggregate WE simulation cost per path: ≈ 2.0 μs.

Path 1 is the dominant exit route, with $k_{\text{off}} = 1.153 \times 10^3$ s^{-1}, approaching the experimental value of 600 ± 300 s^{-1}; the mean is inflated by one high replicate, while the remaining two replicates (~ 740 s^{-1}) fall within experimental uncertainty. Path 2 contributes a secondary rate of 46.5 s^{-1} ($\approx 4\%$ of the total flux), and Path 3 is essentially non-contributing ($< 0.1\%$). These results indicate that Path 1 dominates the reactive flux by more than an order of magnitude, consistent with the M-WEM finding that accurate kinetic estimates can be obtained by considering only the dominant pathway [**Ray2021**]. At the same time, the framework quantifies the mechanistic heterogeneity that would be invisible to single-pathway methods.

Beyond kinetic characterisation, the PathGennie trajectory ensemble provides mechanistic insight into the dissociation mechanism. Along Path 1, we identified a transition-state-like configuration at the point of maximum effort (i.e. maximum N_{trial} required); this configuration is consistent with the findings of Wolf et al. using dissipation-corrected targeted MD (dcTMD) [**Wolf2020**]. Furthermore, a transient hydrogen bond between the N2 atom of benzamidine and Gly212 was observed during dissociation along Path 1 (Fig. 5.11), stabilising an intermediate along the dominant channel. A similar interaction has been documented independently using the site identification by ligand competitive saturation (SILCS) method [**Yu2025**].

Table 5.4 places the WE-PathCV result in the context of the broader literature. The path-resolved framework achieves a rate consistent with experiment at a simulation cost of ≈ 2.0 μs per pathway-specific WE run, comparable to or more efficient than many established approaches, while additionally resolving the mechanistic heterogeneity of the dissociation process.

Method	k_{off} (s^{-1})	Simulation time (μs)
Experiment [**Guillain1970**]	600	—
M-WEM [**Ray2021**]	791 ± 197	0.48
dcTMD [**Wolf2020**]	270 ± 40	0.80
InMetaD & VAC-MetaD [**Brotzakis2018**]	4176 ± 324	~ 1.00
$OPES_f$ [**Ansari2022**]	1560	3.2
MMVT SEEKR [**Jagger2020**]	174 ± 9	4.40
SEEKR2 [**Votapka2022**]	990 ± 130	5.00
LiGaMD [**Miao2020**]	3.53 ± 1.41	5.00
WExplore [**Donyapour2019**]	840	8.75
REVO [**Donyapour2019**]	266	8.75
WE-PathCV Path 1 (this work)	1153	≈ 2.0

Table 5.4: Comparison of k_{off} estimates and aggregate simulation costs for benzamidine–trypsin dissociation from various computational approaches.

5.11 Discussion

PathGennie and the associated path-resolved kinetics workflow address a methodological gap that has persisted in the rare-event simulation community: the disconnect between rapid path discovery and quantitative kinetic resolution. Existing methods tend to excel at one task or the other. CV-biasing methods can yield accurate free energies but require prior knowledge of the coordinate and do not naturally decompose kinetics by pathway. Path sampling methods focus on the transition ensemble but require initial paths or interfaces. Standard WE methods yield unbiased kinetics but can be inefficient when initialised without prior knowledge of the transition region, and they can suppress mechanistic diversity through trajectory merging. PathGennie bridges these domains by providing a computationally inexpensive mechanism for generating initial paths that seed more rigorous kinetic calculations, and the path-resolved WE framework converts the resulting pathway classification into quantitative kinetic observables.

The central output of the framework is not simply a set of trajectories or clusters, but pathway-specific steady-state fluxes and their relative contributions. This distinction is important for systems such as Abl-imatinib, where a geometrically common route need not carry the dominant kinetic flux. The population-vs-flux distinction demonstrated for the N368S mutation provides a compelling example: a highly sampled channel can be kinetically subdominant, and only WE-based rate estimation can resolve this distinction. For drug resistance studies, relative flux redistribution may be a more statistically reliable design observable than an absolute residence time, because the direction of the crossover is robust across replicates even when absolute rates have broad confidence intervals.

The iterative refinement protocol demonstrated on OAMe-G2 addresses a related challenge: when the initial progress variable cannot discriminate competing mechanisms (here, wet versus dry unbinding), the refinement loop provides a systematic route from

a crude geometric coordinate to a physically validated, mechanism-specific PathCV. The convergence of the Fréchet distance between successive refined paths offers a transparent and interpretable stopping criterion. The subsequent WT-MetaD validation, which recovers a free-energy surface consistent with the brute-force ZTS reference, establishes that the iteratively refined PathCV has practical utility beyond exploratory path generation.

The PathGennie design reflects a deliberate balance between three competing objectives: (1) physical realism, maintained by using unbiased MD equations of motion; (2) computational efficiency, achieved by selective propagation of productive segments; and (3) robustness to coordinate misspecification, enabled by the runner segment that permits relaxation in unguided degrees of freedom. The Wolfe-Quapp benchmark demonstrates that even with an incomplete progress variable, PathGennie can recover pathway diversity when τ_2 is sufficiently long.

Several limitations deserve explicit statement. First, pathway completeness is inherited from the initial PathGennie trajectory ensemble: neither the trajectory count nor the clustering procedure provides a principled stopping rule showing that all significant channels have been identified. Second, non-orthogonal tube definitions can lead to channel switching and loss of pathway identity; the additional z-bins used for the trypsin system are a system-specific mitigation, not a general solution. Strongly overlapping pathways require history-labelled bins or a joint multi-channel WE calculation. Third, the present estimates have not been subject to systematic force-field sensitivity tests; rates and pathway populations may depend on the force field and solvent model. Fourth, pre-seeding changes the initial distribution but not the asymptotic WE estimator; each system therefore requires evidence that the initial transient has decayed before the analysis window, and the current window selection remains partly inspection-based.

Methodologically, these limitations motivate joint channel-labelled WE simulations and hierarchical adaptive MSM (haMSM) or related non-Markovian post-processing for earlier and more objective rate estimation [**Copperman2020**]. Adaptive or binless resampling schemes could replace fixed bins in difficult channels [**Torrillo2021**, **Mitra2025**, **Shahid2026**], while optimised trajectory management may reduce replicate variance [**Bose2025**]. Beyond ligand dissociation, the same flux-decomposition principle can be applied to competing folding and conformational-switching mechanisms.

5.12 Conclusions

This chapter presented a two-stage computational framework developed across two successive studies. The first stage, PathGennie, is a direction-guided adaptive sampling algorithm that generates rare-event transition pathways at computational costs orders

of magnitude below conventional MD. By selectively propagating short, unbiased trajectory segments that demonstrate progress along a defined coordinate, PathGennie rapidly crosses free-energy barriers while preserving physical realism. The algorithm successfully resolved exit routes for benzene in T4 lysozyme, imatinib in Abl kinase, and folding/unfolding trajectories for fast-folding proteins, with wall times ranging from minutes to tens of minutes on a single GPU. Robustness to coordinate misspecification was demonstrated on the Wolfe-Quapp potential.

The second stage, the path-resolved WE framework, converts the raw path ensemble from PathGennie into quantitative, route-specific kinetic information. By clustering path ensembles with DTW, iteratively refining representative curves with neural networks until convergence (monitored by the Fréchet distance), constructing normalised PathCVs, and running pre-seeded independent WE simulations along each PathCV, the framework decomposes global rate constants into route-specific contributions. Path-informed WE achieved rate convergence 3–4× faster than standard source-only initialisation for alanine dipeptide and chignolin, effectively eliminating the transient discovery phase.

Applications demonstrated the power of the combined approach. On the three-hole model potential, path-resolved WE correctly captured a temperature-dependent crossover between competing kinetic channels that is invisible to single-coordinate WE. For the OAMe-G2 host-guest system, iterative path refinement from a simple geometric CV converged onto a validated, mechanism-specific wet-unbinding PathCV in seven iterations. For imatinib dissociation from Abl kinase, the framework revealed that the N368S resistance mutation does not simply accelerate unbinding uniformly but fundamentally redirects reactive flux from the P-loop/hinge channel to the αC-helix channel, providing a mechanistic rationalisation of the experimentally observed threefold acceleration of imatinib release. For benzamidine–trypsin, the framework recovered a k_{off} consistent with the experimental value at a simulation cost of ≈ 2.0 μs, while simultaneously identifying mechanistic intermediates including a transition-state-like configuration and a transient Gly212 hydrogen bond along the dominant dissociation pathway.

Taken together, these results establish that the combination of PathGennie path discovery, iterative PathCV refinement, and pre-seeded WE kinetics provides a practical and powerful bridge between exploratory pathway generation and quantitative rare-event kinetics. The software is openly available at https://github.com/TeamSuman/PathGennie/releases/tag/v1.1.0.

6

TRAILS-MD: Lineage-Aware Adaptive Sampling for the Mapping of Complex Conformational Landscapes and Transition Pathways

6.1 Introduction

Molecular dynamics (MD) simulations constitute a direct and atomistically resolved computational connection between molecular framework and thermodynamic properties and kinetic rates, as described in Chapter 1. By numerical integration of Newton's equations of motion on empirical potential energy surfaces, MD trajectories expose the microscopic mechanisms that govern a wide range of biomolecular processes such as protein folding [**Dill2012**, **LindorffLarsen2011**], ligand binding and dissociation [**Shaw2010**, **Dror2012**], conformational changes, enzyme catalysis [**Warshel2006**, **Hummer2004**], membrane transport, and phase transitions [**Frenkel2001**, **Alder1957**, **Rahman1964**]. In this sense, MD functions as a computational microscope that complements experimental observables, such as rates, structural ensembles, population distributions, and response functions, by connecting them to the underlying molecular motions [**Karplus2002**].

The fundamental limitation of all-atom MD lies in the vast separation between the shortest dynamical timescale that must be resolved and the slow, rare events that are chemically and biologically important. Stable numerical integration of bonded motions requires a timestep of approximately 1–2 fs to adequately sample hydrogen-atom vibrations [**Leach2001**], whereas activated conformational transitions typically occur on microsecond, millisecond, or even longer timescales.

Because transition rates decrease exponentially with the height of the free-energy barrier, as described by Transition State Theory (TST) [**Eyring1935**]:

$$k = \kappa \frac{k_B T}{h} \exp\left(-\frac{\Delta G^{\ddagger}}{k_B T}\right), \tag{6.1}$$

where κ is the transmission coefficient and $\Delta G^{\ddagger}$ is the activation free energy, sufficiently large barriers can render rare events effectively inaccessible on conventional

simulation timescales [**Hnggi1990**, **Berezhkovskii2011**]. Unbiased MD trajectories therefore spend the overwhelming majority of their computational effort fluctuating within metastable free-energy basins, while only rarely crossing the transition regions that govern molecular mechanism and kinetics [**Chandler1987**].

A broad family of enhanced sampling methods has been developed to address this rare-event problem, as surveyed in Chapter 2. CV-biasing methods, such as umbrella sampling [**Torrie1977**, **Kumar1992**], metadynamics [**Laio2002**, **Barducci2008**], adaptive biasing force (ABF) [**Darve2001**, **Hnin2004**] and on-the-fly probability enhanced sampling (OPES) [**Invernizzi2020**], accelerate the conformational exploration by modifying the system Hamiltonian along a small set of selected slow degrees of freedom. These methods have been shown to be pivotal to free-energy calculations, but their accuracy is intrinsically related to the quality of the chosen collective variables. Poorly specified coordinates that neglect important slow degrees of freedom can lead to hysteresis, slow orthogonal relaxation, and systematically wrong free-energy profiles [**Rohrdanz2011**, **Bonati2021**].

Generalized-ensemble techniques such as parallel tempering and Hamiltonian replica exchange [**Sugita1999**, **Fukunishi2002**, **Swendsen1986**] speed up barrier-crossing rates by linking simulations at different thermodynamic states, but their computational cost grows quickly with system size because of the number of replicas required to achieve adequate exchange probabilities [**Sindhikara2008**]. Trajectory-ensemble and path methods such as transition path sampling [**Bolhuis2002**, **Bolhuis2000**], transition-interface sampling [**vanErp2003**], forward flux sampling [**Allen2006**], multiscale milestoning [**Votapka2017**, **Votapka2022**], and weighted ensemble (WE) sampling [**Huber1996**, **Zhang2010**, **Zwier2015**] operate on ensembles of dynamical trajectories. These methods preserve the underlying physical dynamics, but require careful management of statistical weights, trajectory histories, or interface crossings, and can be inefficient in high-dimensional spaces with multiple competing pathways [**Zuckerman2010**]. The path-ensemble perspective is developed further in Chapter 5.

Adaptive sampling provides a complementary strategy that sidesteps many of these difficulties. Rather than committing the entire computational budget to a single long trajectory, adaptive methods run many shorter trajectories, analyze the accumulated data, and launch new simulations from configurations selected according to a prescribed exploration objective. This objective may be to improve coverage of configuration space, reduce uncertainty in a kinetic model, enrich the sampling of transition pathways, or bias walkers toward a defined target state. The iterative and parallel structure of adaptive sampling has been exploited in supervised MD approaches [**Deganutti2020**], MSM-guided sampling workflows [**MSM2014**, **Chodera2014**, **Prinz2011**], FAST-style sampling [**Zimmerman2015**], multi-armed bandit adaptive protocols [**Prez2020**], reward-based

adaptive sampling [**Shamsi2018**], high-throughput MD platforms [**Doerr2016**, **Lee2019**], and WE-based rare-event calculations [**Zwier2015**, **Aristoff2018**].

The effectiveness of any adaptive protocol depends on three tightly coupled design choices: the representation used to characterize molecular configurations, the rule used to select new starting points, and the bookkeeping infrastructure needed to preserve the statistical interpretation of the resulting trajectory ensemble. The challenge of constructing informative molecular representations is taken up in Chapter 4, where self-supervised learning approaches are developed for discriminating structurally similar condensed-phase environments.

Simple physical descriptors, such as interatomic distances, backbone torsions, root-mean-square deviations, contact fractions, and radii of gyration, are computationally inexpensive and physically interpretable, but they demand prior mechanistic knowledge that may be unavailable. Data-driven approaches attempt to circumvent this limitation by learning informative coordinates directly from accumulated trajectory data. Principal component analysis (PCA) [**PCA2002**] finds directions of maximum configurational variance, while time-lagged independent component analysis (TICA) and VAMP find slowly decorrelating dynamical modes through variational principles [**No2013**, **PerezHernandez2013tICA**, **Wu2020VAMP**]. Neural-network approaches, including autoencoders, time-lagged autoencoders, VAMPnets, and Deep-TICA, extend these ideas to nonlinear latent embeddings suitable for biasing or spawning decisions [**Wehmeyer2018**, **Hernndez2018**, **Mardt2018**, **Bonati2021**, **Noe2020**]. These data-driven representations reduce reliance on hand-crafted coordinates, but introduce additional practical challenges regarding model retraining, coordinate consistency across adaptive rounds, and restart reproducibility.

Several established frameworks address different parts of this problem. WESTPA 2.0 provides a statistically rigorous environment for weighted ensemble simulations and rare-event rate estimation [**Russo2022WESTPA2**]. FAST accelerates conformational exploration through objective-driven adaptive sampling, generating datasets well suited to subsequent MSM construction [**Zimmerman2015**]. HTMD provides a unified framework for high-throughput simulation and Markov state model analysis [**Doerr2016**]. Workflow managers such as ExTASY [**Hruska_2020**] and DeepDriveMD [**Lee2019**] orchestrate large HPC campaigns by coupling adaptive sampling with machine-learning models. Specialized packages including OpenPathSampling [**Swenson_2018**] and cFFS [**DeFever_2019**] focus on transition-path ensembles and rate calculations using dedicated statistical formalisms.

This chapter presents *TRAILS-MD* (Trajectory-Restartable Adaptive Iterative Lineage-Sampled Molecular Dynamics), a lightweight and engine-agnostic adaptive sampling framework that combines short parallel trajectory propagation, fixed or on-the-fly machine-learned collective-variable spaces, interchangeable spawning policies,

restartable workflow state, and, critically, explicit parent–child trajectory lineage tracking. A defining feature of the approach is that every spawned walker stores its ancestry, enabling traceable reconstruction of continuous structural pathways from the output of highly parallelized, disjointed exploration stages. This design distinction is essential: configurational coverage metrics reveal *where* walkers have been, but trajectory lineage determines *whether* the sampled structures can be assembled into a dynamically connected and mechanistically meaningful pathway. We demonstrate the efficacy of TRAILS-MD across a hierarchy of molecular challenges, from low-dimensional conformational barriers in peptide isomerization, through entropic bottlenecks in protein folding, to the rapid mapping of transient ligand-exit tunnels, and show how lineage-aware exploratory ensembles can serve as a practical starting point for quantitative Markov state model (MSM) construction.

6.2 The TRAILS-MD Framework

6.2.1 Overall Algorithm

TRAILS-MD operates as an iterative active-learning workflow in which short parallel MD trajectories are generated, projected into a chosen sampling space, analyzed to select new starting configurations, and then re-launched. Rather than running a single continuous trajectory, the algorithm distributes the computational budget across an ensemble of walkers, short trajectory segments whose starting points are updated at each iteration based on the accumulated sampling history. This structure naturally enables parallel execution across multiple GPUs and improves sampling diversity compared to a single serial run.

The complete algorithmic workflow is given below. At iteration k, the walker ensemble generates a collection of trajectory fragments:

$$\mathcal{T}^{(k)} = \left\{\tau_1^{(k)}, \tau_2^{(k)}, \ldots, \tau_M^{(k)}\right\}, \tag{6.2}$$

where each fragment $\tau_i^{(k)} = \{\mathbf{x}_{i,0}^{(k)}, \mathbf{x}_{i,1}^{(k)}, \ldots\}$ is the trajectory initialized from the i-th walker in the ensemble $\mathcal{W}^{(k)}$. Frames from $\mathcal{T}^{(k)}$ are projected into the chosen sampling space, scored together with the accumulated history, and mapped back to full-system coordinates to define the next ensemble $\mathcal{W}^{(k+1)}$.

An important design choice concerns the treatment of velocities at spawn points. TRAILS-MD extracts only spatial coordinates and periodic box vectors from selected parent frames; velocities are not inherited but are instead redrawn from a Maxwell–Boltzmann distribution at the target temperature before each new segment begins. This choice deliberately breaks exact phase-space continuity in order to enhance rapid exploration and decorrelation between walker histories. Consequently, the short walker

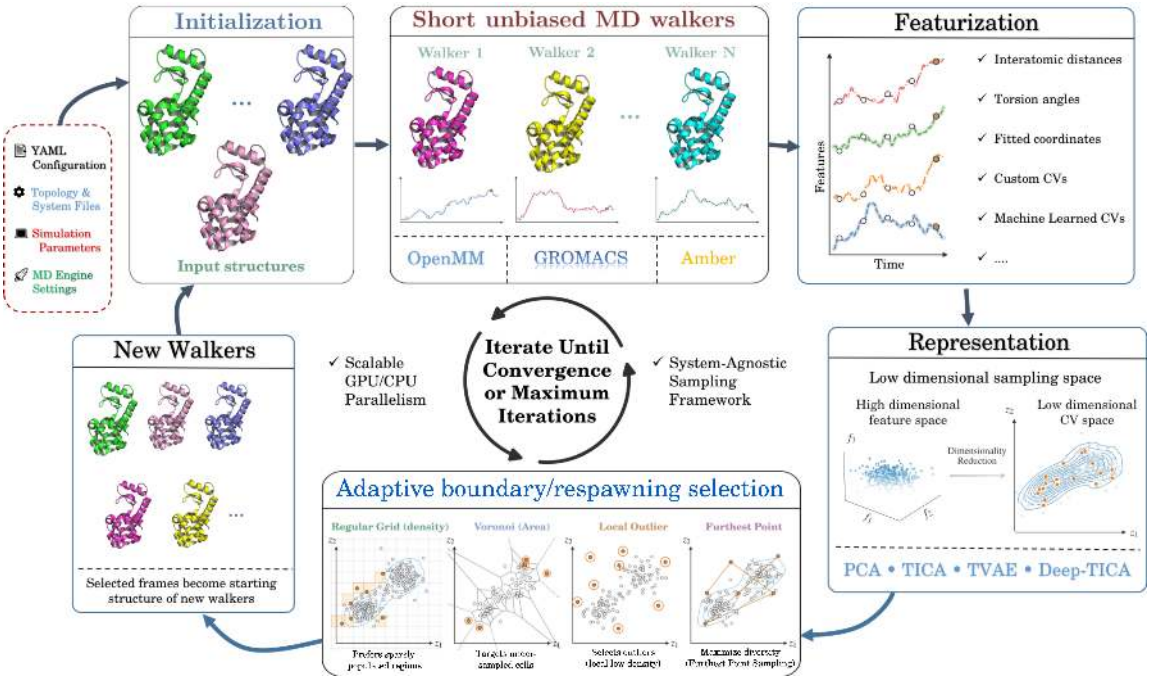

Figure 6.1: Schematic workflow of TRAILS-MD. A YAML configuration file defines the molecular system, MD engine, sampling space, spawning rule, and checkpointing options. The core driver prepares the requested engine and launches an ensemble of short parallel walker trajectories. Generated frames are converted into fixed collective variables or molecular features; in adaptive mode, PCA, TICA, TVAE, or Deep-TICA models are trained or updated at scheduled intervals, and historical features are reprojected when the latent representation changes. Spawning is performed on the accumulated projected history, after which the selected frames are mapped back to full-system coordinates to initialize the next walker ensemble. At each iteration, the framework records trajectory history, parent–child lineage, checkpoint state, occupancy diagnostics, and run logs, enabling restartable and fully traceable adaptive sampling campaigns.

trajectories do not form an unbiased thermodynamic or kinetic ensemble. For rigorous kinetic calculations, the exploratory structures identified by TRAILS-MD should be used to seed longer, continuous production runs, as demonstrated in the MSM construction application described in Section 6.4.6. The overall workflow is illustrated schematically in Fig. 6.1 and in the algorithmic description below.

Algorithm 1: TRAILS-MD workflow.

Input : Initial structures, topology, MD engine, sampling space definition, spawning rule, number of walkers M, maximum iterations K

Output: Trajectory ensemble, projected coordinates, lineage records, occupancy history

Initialize walker ensemble $\mathcal{W}^{(0)}$ from supplied starting coordinates
for $k = 0, 1, \ldots, K-1$ **do**
    Propagate: Run M short MD trajectories from walkers $\mathcal{W}^{(k)}$
    Represent: Extract features or fixed CVs from generated frames
    if *scheduled* **then**
        Fit or update adaptive model on accumulated features; reproject entire history into updated space
    Project: Map current frames to sampling coordinates
    Store: Write CVs, optional features, and iteration metadata
    Spawn: Score cumulative history and select M parent frames
    Extract: Recover full-system coordinates of selected parent frames
    Link: Record parent–child trajectory lineage
    Checkpoint: Save restart state and sampling history
    Assess: Update occupancy and convergence diagnostics
    Set $\mathcal{W}^{(k+1)}$ to the extracted spawn structures

6.2.2 Trajectory Propagation

The trajectory generation layer provides a unified interface to multiple MD engines. TRAILS-MD currently supports OpenMM (accessed directly through its Python API), GROMACS, and Amber (both launched as external executables with per-walker input files generated on-the-fly). For GPU-accelerated runs, each worker is assigned to a specified compute device, allowing parallel walkers to be distributed across all available hardware.

Each engine implements two fundamental operations: preparation of the simulation environment from the supplied topology and starting coordinates, and production of one simulation segment from a selected starting structure. If a production segment fails, for example due to a numerical instability, TRAILS-MD removes the incomplete output, attempts a clean restart by rebuilding the simulation state, performing a short energy minimization, and when requested, carrying out a brief re-equilibration. If the

retry also fails, the iteration is halted before analysis begins, ensuring that incomplete trajectory data never enters the adaptive sampling history and corrupts subsequent spawning decisions.

6.2.3 Featurization and Coordinate Representation

Raw Cartesian coordinates are sensitive to rigid-body translations and rotations and are therefore unsuitable as input for dimensionality-reduction models. TRAILS-MD converts molecular configurations into rotation- and translation-invariant feature vectors before projection. The supported feature types include:

1. **Pairwise distances:** For a selection of N atoms, all unique interatomic distances are computed:
$$d_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|, \quad \forall\, i < j, \tag{6.3}$$
yielding a feature vector of dimension $N(N-1)/2$.

2. **Aligned Cartesian coordinates:** Configurations are superimposed onto a reference structure via least-squares fitting, and the aligned atom positions are used directly:
$$\mathbf{x} = \left[\mathbf{r}'_1, \mathbf{r}'_2, \ldots, \mathbf{r}'_K\right], \quad \mathbf{r}'_i \in \mathbb{R}^3. \tag{6.4}$$

3. **Fixed physical CVs:** User-defined coordinates such as backbone dihedral angles, RMSD from a reference structure, radius of gyration, fraction of native contacts, or ligand center-of-mass displacement are computed directly from trajectory frames without dimensionality reduction.

The resulting descriptor vectors are collected into a cumulative dataset $\mathcal{X}^{(k)} = \{\mathbf{x}_1, \ldots, \mathbf{x}_P\}$ that grows with each iteration. In adaptive mode, this dataset serves as the training set for the projection model.

To avoid excessive retraining and sensitivity to fluctuations in early sampling rounds, adaptive models are updated only at a pre-defined frequency. When a model is retrained, all previously stored features are reprojected through the updated model, ensuring that the complete sampling history is represented in a consistent coordinate system before spawning decisions are made.

6.2.4 Sampling-Space Projection Models

The accumulated feature dataset $\mathcal{X}^{(k)}$ is mapped into a low-dimensional sampling space:

$$\mathbf{z} = f(\mathbf{x}), \quad \mathbf{z} \in \mathbb{R}^m, \tag{6.5}$$

where m is the target dimensionality (typically 2 or 3). TRAILS-MD supports four adaptive projection models as well as fixed physical CVs.

Principal Component Analysis (PCA)

PCA provides a linear projection that identifies orthogonal directions of maximum variance in the feature space. For a centered feature matrix $\mathbf{X}$, PCA is the solution of the eigenvalue problem

$$\mathbf{C}\,\mathbf{w}_i = \lambda_i\,\mathbf{w}_i, \qquad \mathbf{C} = \frac{1}{N-1}\,\mathbf{X}^{\mathsf{T}}\mathbf{X}, \tag{6.6}$$

with $\mathbf{C}$ the sample covariance matrix, λ_i the eigenvalues ordered in decreasing magnitude, and $\mathbf{w}_i$ the corresponding eigenvectors (principal components). Projection onto the leading components yields the low-dimensional representation. PCA serves as an interpretable linear baseline, particularly suited to systems where the dominant conformational motions are approximately linear in the chosen features.

Time-Lagged Independent Component Analysis (TICA)

TICA finds linear combinations of features that are maximally autocorrelated at a user-specified lag time τ, and thus identifies coordinates associated with the slowest processes in the dynamic. It solves the generalized eigenvalue problem [**No2013**, **PerezHernandez2013tICA**]:

$$\mathbf{C}_{0\tau}\,\mathbf{w}_i = \lambda_i\,\mathbf{C}_{00}\,\mathbf{w}_i, \tag{6.7}$$

where $\mathbf{C}_{00}$ is the instantaneous covariance matrix and $\mathbf{C}_{0\tau}$ is the time-lagged cross-covariance matrix evaluated at lag τ. Under the assumptions of the linear TICA approximation, the components corresponding to the largest eigenvalues λ_i approximate the most slowly decorrelating collective modes and therefore provide useful candidate coordinates for adaptive sampling of conformational transitions. In contrast to PCA, which emphasizes high-variance directions, TICA explicitly prioritizes slow dynamical relaxation.

Time-Lagged Variational Autoencoder (TVAE)

The time-lagged variational autoencoder extends the TICA philosophy to a nonlinear latent space. An encoder network maps the configuration at time t to a probability distribution over latent variables:

$$q_\phi(\mathbf{z}|\mathbf{x}_t) = \mathcal{N}\Big(\boldsymbol{\mu}_\phi(\mathbf{x}_t),\ \sigma^2_\phi(\mathbf{x}_t)\,\mathbf{I}\Big)\,, \tag{6.8}$$

while a decoder network is trained to reconstruct the configuration at a later time $t+\tau$, denoted $p_\theta(\mathbf{x}_{t+\tau}|\mathbf{z})$. The model is trained by minimizing the time-lagged evidence lower

bound (ELBO):

$$\mathcal{L} = \mathbb{E}_{q_\phi}\left[-\log p_\theta(\mathbf{x}_{t+\tau}|\mathbf{z})\right] + \beta\, D_{\mathrm{KL}}\left[q_\phi(\mathbf{z}|\mathbf{x}_t) \parallel p(\mathbf{z})\right], \tag{6.9}$$

where the first term penalizes time-lagged reconstruction error and the second term, scaled by the hyperparameter β, regularizes the latent distribution towards a standard Gaussian prior. The time-lag constraint encourages the latent dimensions to encode features of the molecular conformation that remain predictive over the selected lag time, reducing sensitivity to fast local fluctuations. The encoder architecture, dropout, learning rate, lag time τ, batch size, and training epochs are all specified externally through the configuration file.

Deep-TICA

Deep-TICA replaces the linear TICA map with a deep neural network:

$$\mathbf{z}_t = f_\theta(\mathbf{x}_t), \quad \mathbf{z}_t \in \mathbb{R}^d, \tag{6.10}$$

where the network is optimized using a time-lagged variational objective so that the learned coordinates approximate the slowest eigenfunctions of the transfer operator. In TRAILS-MD, Deep-TICA takes pairwise Cα distances as input and produces a latent projection designed to separate metastable conformational states. The network is retrained at prescribed intervals, and all historical features are reprojected through the updated model after each retraining event. Deep-TICA training is performed through MLColvar and PyTorch Lightning, with the network architecture and training hyperparameters supplied in the configuration file [**Bonati2021**, **Bonati2023Mlcolvar**].

The adaptive projection interface is modular: PCA, TICA, TVAE, and Deep-TICA are the currently tested options, but the same interface accommodates other learned coordinates. A new model only needs to specify how accumulated features are fitted and how current and historical frames are projected into a consistent low-dimensional space. This design allows methods such as SPIB, VAMPnets, or Deep-LDA to be incorporated without modifying the main adaptive loop.

6.2.5 Configurational Space Partitioning and Spawning

Once all trajectory frames have been projected into the sampling space, TRAILS-MD uses the resulting point cloud to decide where the next walkers should be initialized. Let $\mathbf{y}_i \in \mathbb{R}^m$ denote the projected coordinate of frame i. At each iteration, the spawning algorithm operates on the *accumulated history* rather than only the most recent trajectories, ensuring that selection pressure is applied to the full explored distribution. Four complementary spawning strategies are implemented.

Density-Based Spawning

Density-based spawning partitions the projected space into a regular rectangular grid. For an occupied bin b containing n_b frames, the exploration weight is:

$$w_b = \frac{1}{n_b}, \tag{6.11}$$

so that bins with fewer accumulated samples are more likely to provide starting points for new walkers. This criterion directly focuses computational effort on sparsely sampled regions, progressively driving the ensemble away from populated basins. When a target state is specified, the density weight can additionally be multiplied by a closeness-to-target factor, balancing exploration of sparse regions with directional progress toward the target. While density grids are highly interpretable and efficient in two or three dimensions, they become impractical for sampling spaces of four or more dimensions due to the exponential growth in the number of bins; geometry-based approaches are preferred for higher-dimensional spaces.

Voronoi Spawning

Voronoi spawning replaces the fixed rectangular grid with a geometry-adaptive partition of the projected space. Cell centers $\{\mathbf{c}_j\}$ are determined by mini-batch k-means clustering of the accumulated projected frames. Each frame is then assigned to the nearest cluster center, defining Voronoi cells:

$$V_i = \left\{ \mathbf{z} \;\middle|\; \|\mathbf{z} - \mathbf{c}_i\| \leq \|\mathbf{z} - \mathbf{c}_j\|,\ \forall\, j \neq i \right\}. \tag{6.12}$$

The spawning weight for cell j accounts for both the sparsity of sampling and the geometric extent of the cell:

$$w_j = \frac{A_j}{n_j}, \tag{6.13}$$

where n_j is the number of frames assigned to the cell and A_j is an estimate of its geometric measure (area in two-dimensional projections, or volume in higher-dimensional projections) obtained by assigning a uniform grid of probe points to their nearest Voronoi center. Because cells at the frontier of the explored manifold are geometrically large relative to their occupancy, Voronoi spawning preferentially places new walkers near the boundaries of the current sampling distribution, driving expansion into unvisited regions of configuration space.

Local Outlier Factor (LOF) Spawning

LOF spawning selects configurations that are locally unusual relative to their immediate neighborhood in the projected space [**Breunig2000**, **Doerr2016**]. For each frame, a local density is estimated from its k nearest neighbors. Frames whose local density is substantially lower than that of their neighbors receive higher outlier scores and therefore larger spawning weights. This criterion is particularly useful when the desired starting points lie at the edges of sampled conformational clusters or along emerging transition channels, where they may not be well captured by grid-based density metrics.

Farthest-Point Sampling (FPS)

Farthest-point sampling provides a deterministic geometric criterion for maximizing the spatial coverage of the projected ensemble. Starting from an initial seed frame, FPS iteratively selects the point farthest from all already-selected frames:

$$i^{\star} = \arg\max_{i} \min_{j \in S} \left\| \mathbf{y}_i - \mathbf{y}_j \right\|_2 , \tag{6.14}$$

where S is the set of frames already selected for spawning. FPS spreads out the new walkers uniformly and widely in the projected space by maximizing the minimum distance to all already chosen points. This strategy is most useful when the primary goal is geometric coverage rather than bias toward a specific low-density bin or target region.

6.2.6 Sampling Coverage and Convergence

At the conclusion of each iteration, TRAILS-MD updates a set of coverage diagnostics by binning the accumulated projected point cloud on a regular grid (or assigning it to the current Voronoi partition) and counting occupied bins. This occupancy count provides a simple and transparent exploration-saturation monitor: if the number of newly occupied bins does not increase for a configurable number of consecutive iterations (`resolution_check_patience`), the grid or Voronoi resolution is automatically increased up to a configured maximum. If the occupancy remains unchanged for a second configurable patience threshold (`convergence_patience`), the adaptive campaign is terminated early and flagged as saturated in the monitored projection.

This occupancy criterion measures *spatial exploration* rather than thermodynamic or kinetic convergence, and is therefore most appropriate for exploratory sampling, pathway discovery, and the identification of representative starting structures for follow-up simulations. For workflows aimed at quantitative kinetic modeling, additional convergence diagnostics, such as implied-timescale stability, Chapman–Kolmogorov tests, or VAMP score convergence, should be incorporated as additional stopping

conditions.

The interpretation of the occupancy diagnostic depends on the sampling space. In a fixed space with predefined bounds, occupancy measures cumulative coverage of a permanent grid and is directly comparable across iterations. In an adaptive space, the coordinate system can rotate, shift, or rescale after model retraining. TRAILS-MD dynamically recalculates grid boundaries in the newly projected space and evaluates the spawning rule on the re-binned history, ensuring that new walkers are always placed based on the most current spatial representation.

6.2.7 Post-Processing and Seeding for Kinetic Analysis

After the adaptive exploration campaign has sufficiently covered the sampling space, the resulting trajectory ensemble can be used in two complementary ways. First, the accumulated conformations may be used to identify or refine collective variables that describe the dominant conformational changes, for example by training a TICA or TVAE model on the full exploratory dataset. Second, the explored space may be discretized to select representative structures as starting points for longer, unbiased production trajectories suitable for quantitative kinetic modeling.

The short adaptive walker trajectories produced by TRAILS-MD are primarily designed for rapid exploration and should not in general be treated as a direct kinetic estimator. Their non-equilibrium initialization, short length, and adaptive selection complicate the assumptions of local equilibration and Markovianity required for standard Markov state model (MSM) construction. For this reason, TRAILS-MD treats adaptive exploration and kinetic estimation as separable and sequential stages: the exploratory stage identifies the important regions of configuration space and produces representative structures distributed across the explored manifold, while the kinetic estimation stage uses these structures to seed longer, less-biased production trajectories. TRAILS-MD supports this two-stage workflow by managing the discretization of the explored space and generating new starting structures from selected regions of the adaptive ensemble.

6.3 Computational Details

All molecular dynamics simulations were performed with OpenMM, GROMACS, or Amber, depending on the molecular system and benchmark protocol [**Eastman2017**, **Abraham2015**, **Pll2015**, **amber2023**]. Unless stated otherwise, each system was energy-minimized and equilibrated before the adaptive sampling campaign was begun. The following subsections summarize the simulation models, collective variables, and TRAILS-MD settings used for each system. Full input files and YAML configuration examples are provided in the software repository.

6.3.1 Alanine Dipeptide in Explicit Water

Alanine dipeptide (ACE–ALA–NME) was used as a fixed-CV reference system for which the relevant conformational coordinates are well established. The system was modeled in explicit TIP3P water using the AMBER99SB force field. PME electrostatics with a 1.0 nm cutoff and an Ewald error tolerance of 5×10^{-4} were applied. Bonds to hydrogen atoms were constrained with a tolerance of 10^{-6}, and hydrogen mass repartitioning used a hydrogen mass of 1.5 amu. Dynamics were propagated using a Langevin integrator at 300 K with a collision frequency of 1 ps^{-1} and a 2 fs timestep.

The TRAILS-MD run used OpenMM on CUDA with mixed precision. Each iteration launched 8 walkers for 1000 steps per walker, corresponding to 2 ps per walker segment. Coordinates were saved every 50 steps. Hard density spawning was applied on a 36×36 grid spanning $\phi, \psi \in [-180°, 180°]$. The backbone dihedral angles ϕ and ψ were used as the sampling space.

6.3.2 Chignolin (CLN025)

Chignolin CLN025 (sequence GYDPETGTWG) was studied using an implicit-solvent OpenMM system with the ff14SBonlysc force field [**Nguyen2014**, **Shao2018**]. Solvent was treated with the HCT generalized Born model [**Nguyen2013**, **Hawkins1996**] with a solute dielectric of 1.0 and a solvent dielectric of 78.5. Dynamics were propagated at 275 K with a Langevin integrator (collision frequency 5 ps^{-1}, timestep 2 fs), using NoCutoff for nonbonded interactions.

Three sampling spaces were compared. The fixed physical CV run used Cα radius of gyration (R_g) and Cα RMSD from the folded reference, with 16 walkers, 5000 steps per walker, and a 30×30 spawning grid spanning R_g = 4–10 Å and RMSD = 0–8 Å. The adaptive TVAE run used Cα pairwise distances as features, a two-dimensional TVAE latent space, a 30×30 grid, 16 walkers, and model retraining every five iterations. The TICA run used the same feature set with a two-dimensional time-lagged projection, 8 walkers, and iteration-level model updates with a lag time of 1 ps.

As an explicit-solvent reference, a CLN025 trajectory using the CHARMM27* protein force field with TIP3P water was generated at 340 K in a cubic box of approximately 4 nm side length, following the DEShaw Research fast-folding benchmark setup [**LindorffLarsen2011**].

6.3.3 AIB9 Peptide

The AIB9 (*N*-methylalanine nonapeptide) system was used to investigate whether adaptive sampling can distinguish independent basin discovery from a genuine dynamical transition between left- and right-handed helical states. The reference topology was generated with the CHARMM-GUI FF-Converter and contained 1540 TIP3P water

molecules and 4749 atoms in total. Simulations were performed with OpenMM on CUDA using NPT dynamics at 400 K and 1 bar with a 2 fs timestep.

Multiple sampling spaces were tested. The baseline DeepTDA–ϕ_5 run used 12 walkers for 2500 steps per walker. The VAE latent-space run used 12 walkers for 5000 steps per walker and a 48×36 grid. The handedness–asymmetry run used a 50×30 grid. All AIB9 runs used hard density spawning and saved trajectory features for lineage analysis.

6.3.4 Trp-Cage and WW Domain Protein Folding

The Trp-cage miniprotein (PDB 2JOF) was simulated using the CHARMM27* force field with TIP3P water and Beglov ion parameters at 290 K. TRAILS-MD was run with 25 walkers, each propagated for 40 ps per iteration, over 750 adaptive iterations. The sampling space was defined by RMSD from the native structure and the fraction of native contacts.

The WW domain benchmark used GROMACS with a CHARMM22*/TIP3P setup based on the DEShaw GTT trajectory dataset [**LindorffLarsen2011**]. The solvated system contained 562 protein atoms (35 residues), 4121 water molecules, 3 chloride ions, and a cubic box of approximately 5.2 nm per side. Energy minimization used steepest descent, followed by NVT (100 ps) and NPT (500 ps) equilibration with position restraints. Production used GPU-offloaded GROMACS with 16 walkers, 5000 steps per walker (10 ps per walker), and density spawning on a 50×50 grid over the fraction of native contacts Q_{native} and R_g. After an unfolding exploration campaign, a TICA model was trained on the generated trajectories and its leading three components used as the sampling space for a subsequent folding campaign.

6.3.5 Ligand Unbinding: OAMe-G2 and HSP90

The OAMe-G2 host–guest complex was adapted from the PLUMED-NEST SAMPL7 benchmark [**Bonomi2019**, **Sun2020**, **Khalak2020**]. The two-dimensional sampling space used the axial ligand center-of-mass projection z and a water-coordination order parameter V_2 to distinguish wet and dry unbinding channels.

For HSP90, the N-terminal domain (PDB 6EI5) with bound inhibitor B5Q [**Musil2017**, **Kokh2018**] was simulated in a periodic box of approximately $7.36 \times 7.30 \times 7.30$ nm^3 containing the protein, one B5Q ligand, nine sodium ions, and 11890 water molecules (39019 atoms in total). The OpenMM setup used PME electrostatics with a 1.0 nm cutoff, hydrogen-bond constraints, rigid water, and hydrogen mass repartitioning to 1.5 amu. Dynamics used a Langevin integrator at 300 K with a 2 fs timestep. TRAILS-MD ran 16 walkers for 5000 steps per walker with four GPU IDs. The three-dimensional sampling space was defined by the Cartesian coordinates of the B5Q center of mass

after alignment to protein Cα atoms, with density spawning on a $30 \times 30 \times 30$ grid.

6.4 Results

6.4.1 Alanine Dipeptide: Validating the Framework with a Known Coordinate

Alanine dipeptide provides an important validation case in which the relevant low-dimensional collective variables, the backbone dihedral angles ϕ and ψ, are known from decades of prior simulation and experimental work. The Ramachandran free-energy landscape contains four commonly discussed conformational regions: C_7^{eq} ($\phi \approx -80°$, $\psi \approx 80°$), α_R ($\phi \approx -60°$, $\psi \approx -40°$), α_L ($\phi \approx 60°$, $\psi \approx 60°$), and C_7^{ax} (included as a diagnostic reference state). These basins are separated by barriers on the order of several $k_{\mathrm{B}}T$.

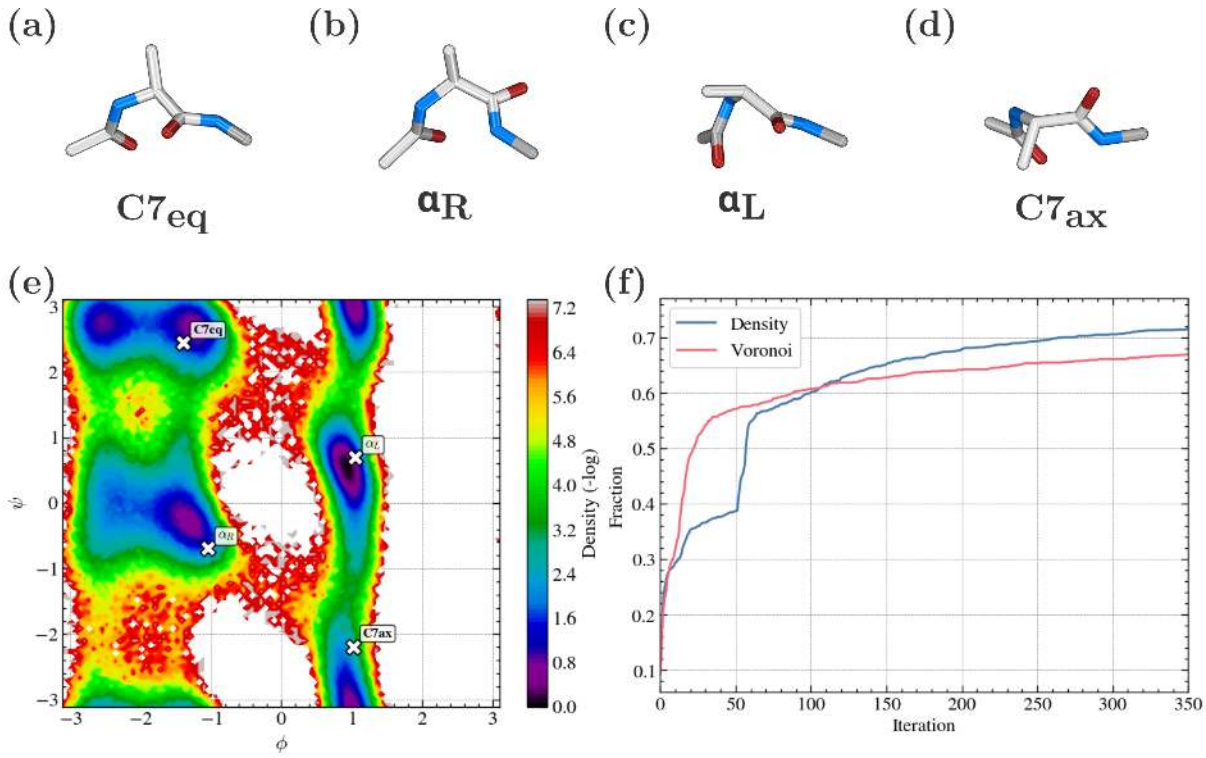

Figure 6.2: Adaptive sampling of alanine dipeptide in the Ramachandran plane. Representative structures corresponding to the (a) C_7^{eq}, (b) α_R, (c) α_L, and (d) C_7^{ax} conformations. (e) Cumulative conformational distribution sampled by TRAILS-MD projected onto the (ϕ, ψ) plane; the color scale reports the negative logarithm of the estimated probability density, and white crosses mark the four reference conformations. (f) Cumulative grid coverage as a function of adaptive iteration for density-based and Voronoi-based spawning, both evaluated on the same 36×36 ϕ/ψ grid.

Unbiased MD of the same cumulative duration remained confined to the C_7^{eq} and α_R basins and did not cross into the α_L region, consistent with the short-timescale

limitation described in the introduction. In contrast, TRAILS-MD with density-based spawning continually redirected walkers from densely populated bins toward sparsely visited portions of the Ramachandran map. All four diagnostic conformational regions were recovered within the adaptive campaign, confirming that the sampled walkers traverse the expected physical landscape rather than filling arbitrary grid cells.

The density-based and Voronoi-based spawning schemes showed comparable early-time behavior. At iteration 99, the density run occupied 777 of 1296 grid cells (60.0% coverage), while the Voronoi run occupied 787 cells (60.7% coverage), as shown in Fig. 6.2(f). The similarity in terminal coverage demonstrates that both strategies are effective for this low-dimensional, well-characterized system. The alanine dipeptide example thus establishes that, when an appropriate low-dimensional coordinate is available, TRAILS-MD achieves rapid and reproducible conformational coverage that is simply not attainable with a single unbiased trajectory of equivalent aggregate length.

6.4.2 Chignolin Folding: Comparing Fixed and Learned Coordinate Spaces

Chignolin (CLN025) is a 10-residue peptide that folds reversibly into a stable β-hairpin, making it one of the most widely used fast-folding benchmarks in the MD literature [**LindorffLarsen2011**]. Unlike alanine dipeptide, CLN025 folding is not naturally described by a single well-established two-dimensional coordinate, making it a valuable test case for comparing physically intuitive CVs with coordinates learned from trajectory data.

Three sampling spaces were compared: the fixed physical space defined by Cα R_g and Cα RMSD from the folded reference; an adaptive TVAE space learned from accumulated Cα pairwise distances; and an adaptive TICA projection of the same feature set. To allow direct structural comparison, all three ensembles were projected onto the same physical R_g–RMSD plane for visualization.

The results, summarized in Fig. 6.3, reveal complementary strengths of the two approaches. The physical R_g–RMSD space provides a transparent structural interpretation and a common basis for comparing runs, but it can collapse structurally distinct CLN025 conformations, for example compact misfolded states and the native hairpin, into overlapping regions. The TVAE and TICA spaces, learned from the accumulated trajectory data, can separate conformations that appear degenerate in the physical projection. This distinction matters for spawning decisions: adaptive coordinates that better resolve conformational states can direct walkers toward genuinely new regions of configuration space rather than repeatedly revisiting structurally similar but numerically distinct configurations.

The key finding is not that learned spaces are universally superior to physical coordi-

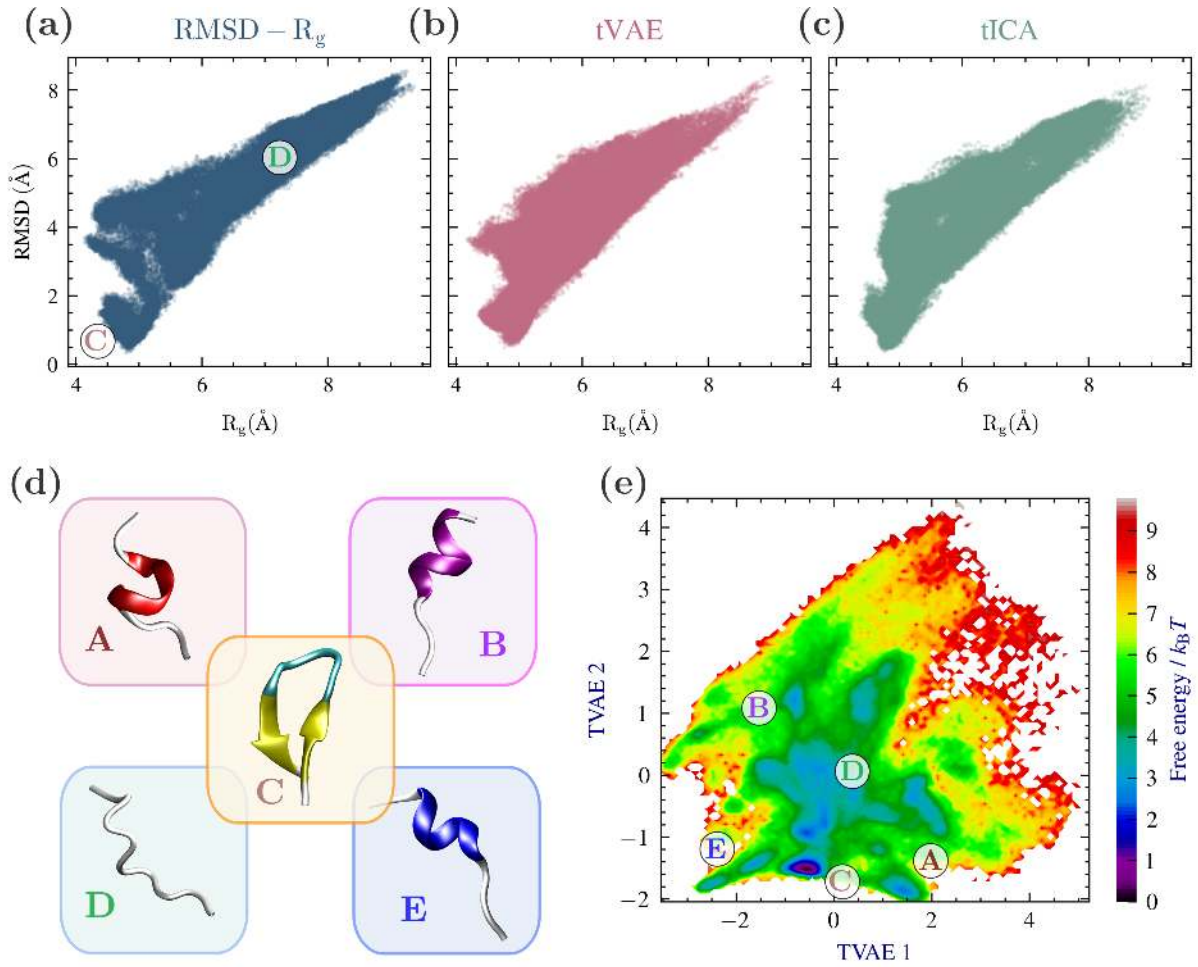

Figure 6.3: Adaptive sampling of CLN025 in physical and learned coordinate spaces. (a) Configurations generated using fixed physical CVs (R_g and RMSD), with spawning performed in the same plane. (b, c) Physical R_g–RMSD projections of configurations generated while spawning in learned two-dimensional TVAE and TICA spaces, respectively. All three panels use the same physical axes to enable direct ensemble comparison. (d) Representative conformations A–E selected from the explored ensemble, spanning compact hairpin-like, partially structured, helical, and extended states. (e) Sampled density in the final TVAE latent space, colored by the relative free-energy-like landscape ($-RT \ln p(\mathbf{x})$) estimated by reweighting with a maximum-likelihood Markov state model rather than directly from the adaptive walker counts.

nates, but that adaptive coordinate learning provides a useful sampling representation precisely when no reliable low-dimensional CV is known before the simulation begins. For systems where the dominant slow coordinates are unclear, learning them from data as the campaign progresses offers a principled and practical alternative to pre-specifying physically motivated observables.

6.4.3 AIB9 Peptide: Basin Discovery Does Not Imply a Pathway

The AIB9 (α-aminoisobutyric acid nonapeptide) system poses a qualitatively more demanding challenge than alanine dipeptide or CLN025. This peptide can adopt both left-handed (L) and right-handed (R) helical states, and the central question is not merely whether both endpoint conformations are discovered, but whether the sampled trajectories establish a *continuous dynamical connection* between them. Resolving this question, which is essential for mechanistic interpretation, requires explicit bookkeeping of which trajectory begat which, rather than relying on spatial coverage alone.

The backbone dihedral angles of AIB9 show qualitatively similar trends across residues, but residues 5 and 8 exhibit the largest barriers in endpoint analysis [**Wang2022**]. When the ϕ_5–ψ_5 plane was used as the sampling space, TRAILS-MD rapidly covered this local torsional subspace; however, the distal dihedral angles remained weakly coupled, and apparent coverage of the sampling space did not correspond to a complete L-to-R helical transition. Expanding the sampling space to the four-dimensional torsional coordinate ($\phi_5, \psi_5, \phi_8, \psi_8$) improved coverage of the monitored subspace but again did not produce a connected L-to-R lineage.

To make the endpoint interpretation more physically transparent, we defined a smooth, signed handedness coordinate. For residue i, the local handedness score is:

$$h_i = \frac{1}{2}\left[\sin(\phi_i) + \sin(\psi_i)\right], \tag{6.15}$$

and the global collective variables are the mean signed handedness:

$$H = \frac{1}{N}\sum_{i=1}^{N} h_i, \tag{6.16}$$

and the terminal asymmetry:

$$A = \frac{1}{3}\sum_{i=N-2}^{N} h_i - \frac{1}{3}\sum_{i=1}^{3} h_i, \tag{6.17}$$

where negative and positive values of H indicate left- and right-handed torsional states, respectively. The asymmetry coordinate A reports end-specific effects and partial helix reversals. This $[H, A]$ representation gives a physically interpretable label for the

endpoint states.

Supervised coordinates trained to discriminate the endpoint states, including Linear Discriminant Analysis (LDA), Time-lagged Discriminant Analysis (TDA), and their deep nonlinear variants Deep-LDA and Deep-TDA [**Bonati2020**], successfully separated labeled L and R examples. TRAILS-MD traversed the corresponding one-dimensional discrimination spaces and, in some runs, reached configurations near both endpoints. However, endpoint proximity in a supervised coordinate space was not sufficient to guarantee pathway connectivity. Several runs discovered both endpoints independently, reached by different root trajectories, without establishing a continuous dynamical connection between them.

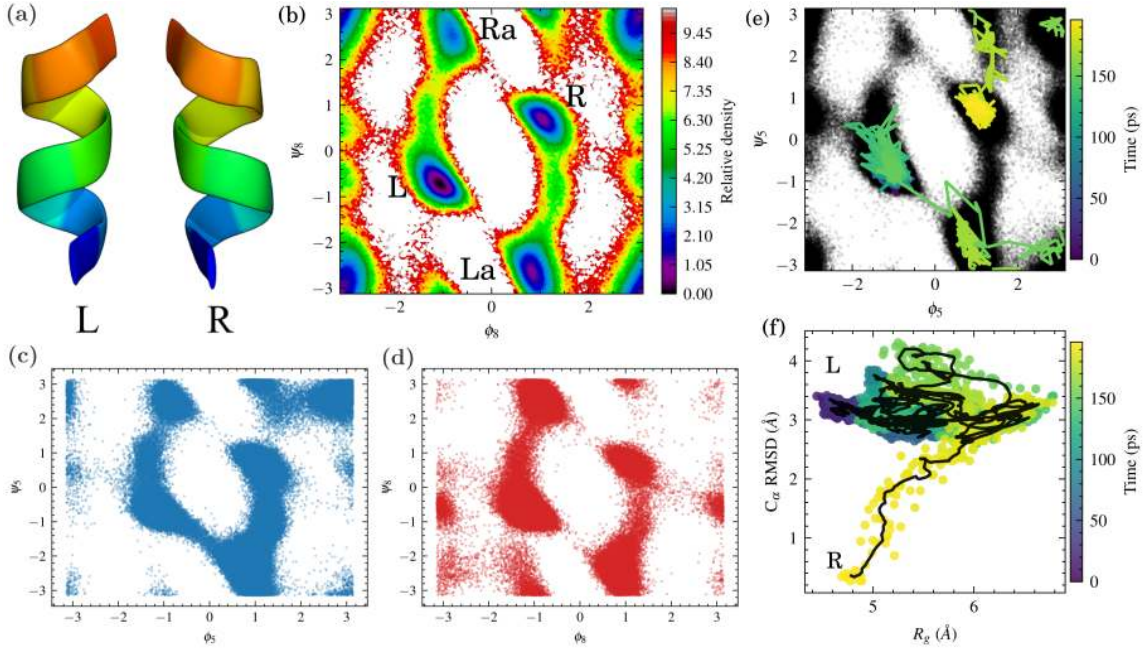

Figure 6.4: Endpoint and torsional analysis of AIB9. (a) Representative left- and right-handed helical conformations of AIB9. (b) Locations of the endpoint conformations in the ϕ_5–ψ_5 torsional projection; the background density-derived landscape is shown as the negative logarithm of the estimated probability density. (c, d) TRAILS-MD sampling in the ϕ_5–ψ_5 and ϕ_8–ψ_8 torsional planes from simulations performed in the four-dimensional torsional space, illustrating endpoint coverage without demonstrating a continuous transition. (e, f) A continuous L-to-R transition trajectory reconstructed by tracing the ancestry of walkers generated in the VAE+LDA projection space, shown in the (ϕ_5, ψ_5) torsional and R_g–RMSD planes (RMSD computed with respect to the right-handed reference R state).

A decisive improvement was obtained by combining an endpoint-discriminating LDA coordinate with a structural VAE latent coordinate. This combined representation improved endpoint specificity and, critically, allowed the reconstruction of a connected L-to-R transition pathway by tracing the parent–child lineage of the resulting walker ensemble (Fig. 6.4(e, f)).

The AIB9 results highlight a central design principle of TRAILS-MD: high spatial coverage, endpoint separation, and pathway connectivity are three *distinct* and non-interchangeable criteria. Because TRAILS-MD stores parent–child relationships throughout the campaign, basin discovery and pathway connectivity can be evaluated independently. This diagnostic capability is critical for any application where the mechanistic question concerns *how* one conformational state is dynamically connected to another, not merely *whether* both states have been visited.

6.4.4 Entropic Bottlenecks in Protein Folding: Trp-Cage and WW Domain

Protein folding presents a qualitatively different challenge from the model systems examined above. Observables commonly used to monitor folding progress, such as RMSD from the native structure or the fraction of native contacts, become increasingly degenerate in the unfolded state, where many structurally distinct conformations map to similar values of the monitoring coordinate. In this regime, a single long trajectory may spend most of its budget diffusing through a large and heterogeneous unfolded ensemble before locating the compact folded basin. An adaptive ensemble of short parallel walkers offers a practical advantage: by redistributing computational effort across diverse regions of configuration space at each iteration, the adaptive protocol increases the probability of reaching the folded state within a given aggregate simulation budget.

To test TRAILS-MD under entropic bottleneck conditions, we considered two fast-folding mini-proteins: the Trp-cage miniprotein (PDB 2JOF, folding timescale ≈ 14 μs) and the WW domain (folding timescale ≈ 21 μs). Both systems provide stringent tests of whether an adaptive ensemble can recover the folded basin from a diverse unfolded starting distribution.

For Trp-cage, we used a physically interpretable two-dimensional sampling space defined by RMSD from the native structure and the fraction of native contacts. With 25 walkers per iteration, each propagated for 40 ps, TRAILS-MD covered a broad region of the projected space over 750 adaptive iterations and reached the folded basin from unfolded initial conditions. The adaptive ensemble explored a substantially larger portion of the physical RMSD–native-contact space than a reference trajectory of comparable aggregate length, consistent with the broader configurational diversity expected from many short parallel walkers.

For the WW domain, we followed a two-stage protocol that exploits the directional asymmetry of folding and unfolding. Because unfolding trajectories are more straightforwardly generated from the native structure, the first stage used TRAILS-MD to explore an unfolding ensemble within approximately 50 adaptive iterations, producing

a diverse set of partially and fully unfolded conformations. A TICA model was then trained on these trajectories, and the leading three time-lagged components were used as the sampling space for a subsequent folding campaign. Starting from unfolded configurations, the folding run explored the learned TICA space efficiently and reached the native basin within approximately 400 iterations out of 500 total.

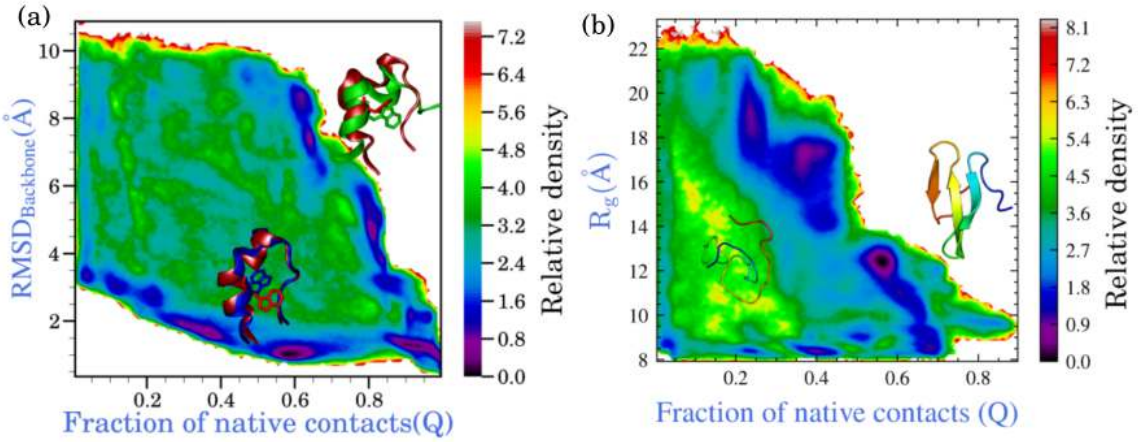

Figure 6.5: Adaptive sampling of two fast-folding proteins. (a) Trp-cage (2JOF) sampled in the RMSD–fraction-of-native-contacts plane, showing exploration from the unfolded ensemble toward the folded basin. (b) WW domain sampled using a three-dimensional TICA space learned from unfolding trajectories and visualized in the physical R_g–fraction-of-native-contacts plane. Representative cartoons are shown within each panel, and relative density is colored by $-RT \ln \hat{p}(\mathbf{x})$.

Figure 6.5 summarizes these two folding applications. The results illustrate a practical and general value of adaptive short-trajectory ensembles for folding problems dominated by entropic broadening. Even when the chosen physical observables are not exact reaction coordinates, they remain useful for monitoring conformational progress and for seeding the next generation of walkers toward less-sampled regions. The two-stage protocol for the WW domain further demonstrates that TRAILS-MD can be composed in a modular way: an initial exploratory campaign defines a data-driven coordinate, which then guides a more targeted folding campaign.

6.4.5 Rapid Mapping of Ligand Unbinding Tunnels

To explore the applicability of TRAILS-MD beyond intramolecular conformational transitions, we applied the framework to ligand unbinding problems. Ligand dissociation presents a complementary challenge: rather than discovering multiple protein conformational basins, the objective is to map the distinct geometrical channels through which the ligand can exit the binding site. We tested two representative cases: the synthetic OAMe-G2 host–guest complex, which is known to exhibit two distinct unbinding routes (commonly described as “wet” and “dry” pathways), and the biologically relevant

N-terminal domain of HSP90 bound to the inhibitor B5Q.

For OAMe-G2, TRAILS-MD progressively expanded sampling away from the bound basin, filling the projected unbinding space across iterations. The coverage fraction of the configured sampling space increased steadily with adaptive iteration, and the distribution of sampled configurations was not confined to a single narrow exit path (Fig. 6.6(b, c)), reflecting the emergence of multiple escape directions consistent with the known dual-channel behavior of this system.

The HSP90 application presents a more challenging protein–ligand environment, where the dissociating inhibitor must navigate a heterogeneous protein surface and locate transient exit tunnels. Unlike the host–guest complex, the challenge here is not purely geometric escape, but the identification of momentarily accessible channels that open and close on simulation timescales. TRAILS-MD explored multiple ligand displacement directions from the initial binding pocket (Fig. 6.6(d, e)), sampling a spatial distribution that outlined several candidate unbinding directions around the protein surface. Rather than converging immediately onto a single dissociation pathway, the adaptive walkers first surveyed multiple displacement modes, which is precisely the behavior required for an initial, exploratory tunnel-mapping stage before more detailed kinetic or mechanistic analysis.

Taken together, these two examples illustrate the breadth of ligand-unbinding applications accessible to TRAILS-MD. In the simpler OAMe-G2 case, the method rapidly identifies multiple exit channels in a compact projected space. In the more complex HSP90 system, it efficiently surveys candidate unbinding tunnels and generates a diverse set of partially dissociated structures for subsequent, more detailed kinetic or mechanistic analysis.

6.4.6 Markov State Model Construction from TRAILS-MD-Seeded Trajectories

The ultimate value of an adaptive exploration framework in a quantitative biophysical workflow depends on whether the exploratory structures can be used to seed rigorous kinetic simulations. To demonstrate this capability, we applied a two-stage TRAILS-MD + MSM strategy to chignolin folding in explicit water, following the CHARMM27*/TIP3P protocol at 340 K from Lindorff-Larsen et al. [**LindorffLarsen2011**].

In the first stage, TRAILS-MD explored the CLN025 conformational landscape in an RMSD–fraction-of-native-contacts (RMSD–FNC) space. After the adaptive coverage reached a stable plateau, the explored space was discretized into a 20×20 grid, and 343 representative spawn-point structures were selected from the occupied regions. In the second stage, each selected structure was used to initialize a 50 ns unbiased production trajectory with CHARMM27*/TIP3P water at 340 K, generating an aggregate

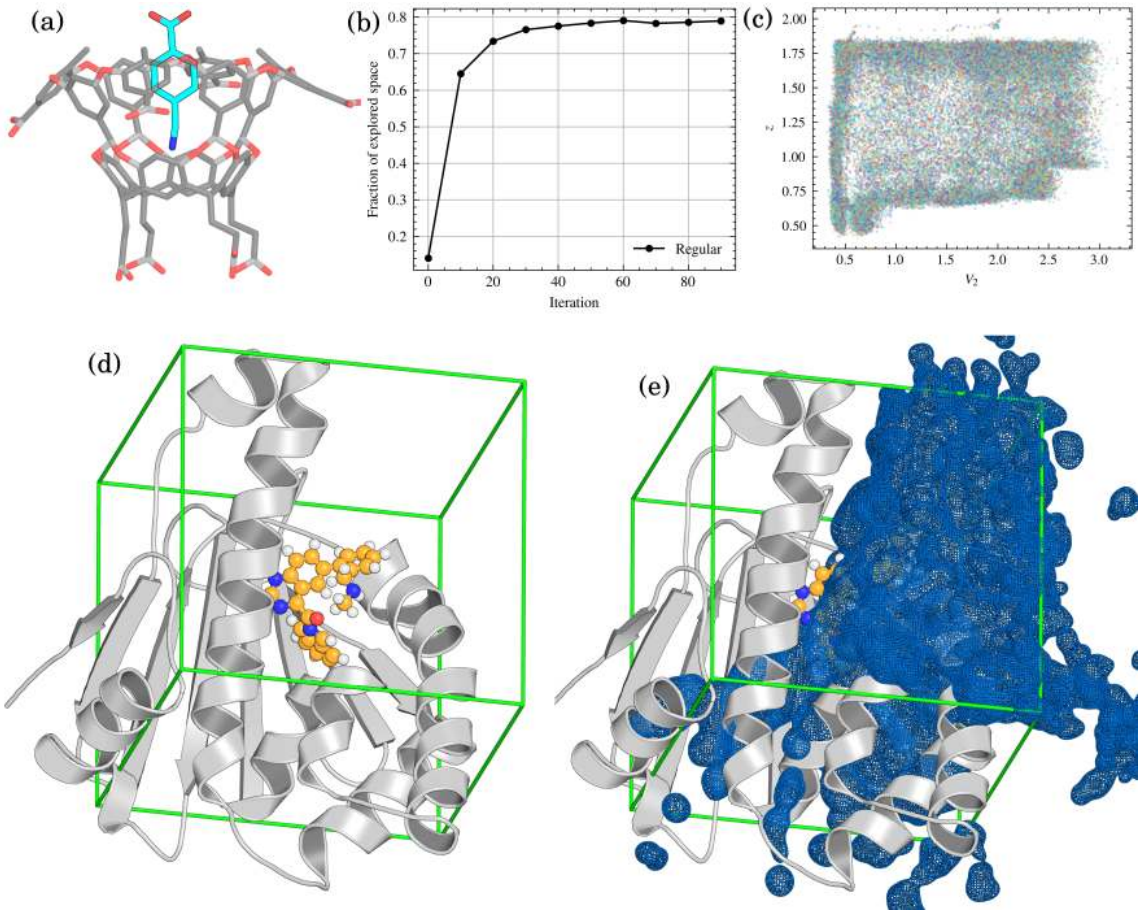

Figure 6.6: Exploratory ligand-unbinding applications of TRAILS-MD. (a) Bound structure of the OAMe-G2 host–guest complex. (b) Fraction of the configured two-dimensional sampling space explored as a function of adaptive iteration for OAMe-G2. (c) Distribution of sampled configurations in the projected host–guest unbinding space, showing the spread of explored positions and emergence of multiple escape directions. (d) Bound N-HSP90α–ligand complex (PDB 6EI5). (e) Superposition of B5Q center-of-mass positions sampled during the TRAILS-MD campaign, providing a qualitative map of possible ligand-exit tunnels around the binding site. The green bounding box denotes the center-of-mass region used to define the TRAILS-MD sampling space.

production dataset of approximately 17.15 μs.

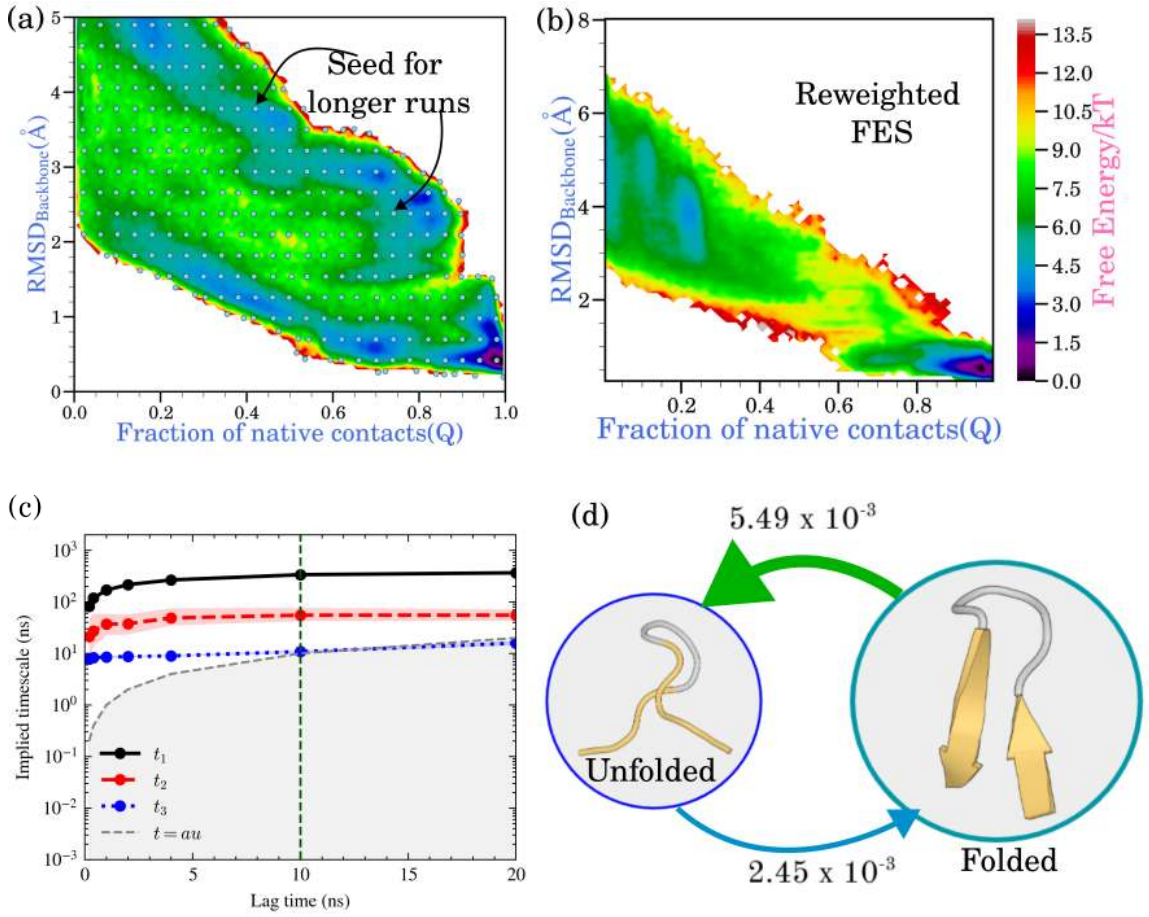

Figure 6.7: MSM construction from TRAILS-MD-seeded trajectories. (a) CLN025 conformational space explored by TRAILS-MD, projected in the RMSD–FNC plane, with selected spawn points marked. (b) MSM-reweighted sampled density in the same physical projection, constructed from production trajectories initialized from the selected spawn structures. (c) Implied timescales as a function of MSM lag time, used to assess Markovian behavior and guide lag-time selection. (d) Two-state coarse-grained MSM with estimated transition probabilities between the folded and unfolded states (illustrated by arrows).

An MSM was constructed from the resulting long trajectories using a lag time of 10 ns, selected based on the convergence of implied timescales (Fig. 6.7(c)). A two-state coarse-grained model recovered folding and unfolding mean first passage times (MFPTs) of approximately 1.6 μs and 0.7 μs, respectively, values consistent with the expected microsecond-scale kinetics of chignolin **[LindorffLarsen2011]**. These estimates were obtained from the long production trajectories, not from the short adaptive walkers themselves; TRAILS-MD served as the systematic distributor of initial structures across the explored landscape, not as the kinetic estimator.

This example demonstrates a practical and general workflow for combining adaptive exploration with quantitative kinetic modeling. Short adaptive walkers rapidly identify

representative conformations across the projected landscape, including regions that would rarely be visited by starting a single long trajectory from the native or unfolded state. Longer unbiased trajectories, seeded from these well-distributed structures, then improve state connectivity, reduce the number of empty transition counts, and provide trajectory data that satisfies the Markovian assumptions required for reliable MSM or related variational kinetic analyses.

6.5 Software, Performance, and Extensibility

6.5.1 Computational Performance and Multi-GPU Scaling

A critical practical requirement for any adaptive sampling orchestration framework is that the analysis and bookkeeping overhead should not dominate the cost of MD propagation. TRAILS-MD is designed to minimize Python-level orchestration costs. Performance was benchmarked using the WW-domain system with 16 parallel walkers, each propagated for 100 ps per iteration, on a four-GPU compute node equipped with NVIDIA GeForce RTX 2080 Ti GPUs running GROMACS 2022.4.

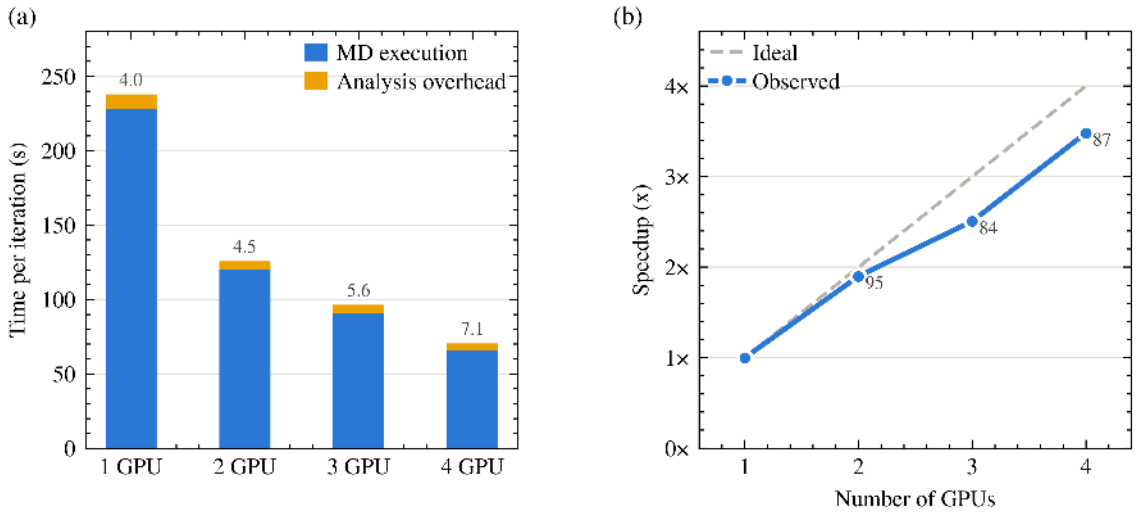

Figure 6.8: Runtime decomposition and multi-GPU scaling of TRAILS-MD for the WW-domain benchmark. (a) Average wall time per adaptive iteration using one to four GPUs, decomposed into MD execution time and non-MD analysis overhead. Numerical annotations above the bars report the overhead fraction of total iteration time in percent. (b) Observed speedup of the MD execution component relative to the one-GPU baseline. The dashed line indicates ideal linear scaling, and values next to markers report corresponding parallel efficiency. Benchmark: 16 parallel walkers of 100 ps each per iteration, GROMACS 2022.4, NVIDIA GeForce RTX 2080 Ti.

The average MD execution time decreased from 228.4 s per iteration with one GPU to 65.6 s per iteration with four GPUs. Over the same range, the non-MD analysis overhead remained between 5.0 and 9.4 s per iteration (Fig. 6.8(a)), corresponding to

4.0% of the total iteration time on one GPU and 7.1% on four GPUs. The observed speedup of the MD execution component reached 3.5-fold on four GPUs, corresponding to a parallel efficiency of 87% (Fig. 6.8(b)). These results demonstrate that the runtime is dominated by MD propagation and that the adaptive orchestration layer introduces only a small and approximately constant overhead.

We also examined the dependence of the analysis overhead on sampling-grid size using two- and three-dimensional TICA spaces, testing grids from 12^2 to 48^3 bins (a 64-fold increase in grid dimensionality). Across this range, the non-MD overhead remained approximately constant per iteration at around 5 s, demonstrating that the occupancy assessment and density-based spawning steps scale gracefully with bin count and do not become a computational bottleneck at the grid resolutions used in practice.

6.5.2 Software Implementation and Dependencies

TRAILS-MD is distributed as a standalone Python package and is available from the project repository at `https://github.com/TeamSuman/Trails-MD`. The framework is designed to be portable across both workstation and high-performance computing environments. GPU, MPI, and job-scheduler support (PBS and SLURM) are delegated to the underlying MD engine; tested backends include GROMACS 2022.4–2026.0, OpenMM 8.0 and later, Amber 2022, and PLUMED-patched variants of these engines, running under CUDA 11.8 and CUDA 12.1.

Standard open-source scientific libraries are used in preference to custom implementations of common algorithms. Conventional machine-learning methods, including PCA, k-means, mini-batch k-means, and LOF, are provided through scikit-learn [**scikit_learn**]. TICA, MSM construction, and related kinetic analyses are performed with Deeptime [**Hoffmann2021**]. Trajectory processing and structural analysis are performed with MDTraj [**McGibbon2015**] and MDAnalysis [**MichaudAgrawal2011**]. Neural network based adaptive models are implemented in PyTorch [**NEURIPS2019_bdbca288**] and Deep-TICA training is performed with the MLColvar library [**Bonati2023Mlcolvar**].

Installation follows the standard Python workflow: after creating an environment with the required MD simulation packages, TRAILS-MD is installed from the repository and executed through the `trails-md` command-line interface. Engine-specific executables, GPU assignments, and run parameters are supplied through a YAML configuration file, allowing the same adaptive workflow to be used across different simulation backends without modifying the core code. The setup should be checked with `trails-md -config config.yaml -check` before running production simulations. Interrupted runs can be resumed with the `-resume` flag, which restores the most recent checkpoint including walker positions, feature history, adaptive model state, and sampling metadata.

6.5.3 Extensibility

TRAILS-MD is designed so that individual components of the adaptive workflow can be replaced without modifying the main iteration loop. New MD engines, spawning strategies, feature definitions, and adaptive sampling spaces are added by implementing the corresponding interface in the `trails_md` package and registering the new component with the internal factory registry. The main loop assumes only that engines generate trajectory files, feature extractors return frame-aligned arrays, spawning methods return global frame indices, and checkpointed objects can be serialized and restored. This modular architecture allows additional approaches, such as SPIB, VAMPnets, DeepLDA, or MSM-guided least-count spawning, to be incorporated with minimal changes to the adaptive workflow.

6.6 Discussion

The examples presented in this chapter were chosen to separate several roles that are often conflated under the broad label of "adaptive sampling." In alanine dipeptide, where the relevant torsional coordinates are known in advance, TRAILS-MD demonstrates its simplest intended use: density-based spawning in a well-chosen low-dimensional space rapidly achieves conformational coverage that is simply not attainable with a single unbiased trajectory of equivalent aggregate length. This case functions as a validation benchmark and illustrates the basic mechanism of the framework.

The CLN025 calculations reveal why this picture becomes more complex for larger systems. Simple physical observables such as R_g and RMSD are useful for visualization and progress monitoring, but they can project structurally distinct conformations onto the same point in the observation plane. Learned spaces, TVAE and TICA, can improve the sampling representation by separating conformations that are degenerate in physical projections. The key insight is not that learned coordinates are universally superior, but that adaptive coordinate discovery during the campaign provides a principled alternative when a reliable low-dimensional CV is unavailable a priori. This theme, learning representations from data rather than prescribing them, connects directly to the self-supervised approach developed for condensed-phase environments in Chapter 4.

The AIB9 example motivates one of the most important design choices of TRAILS-MD: explicit parent–child lineage tracking. Several coordinate choices sampled both left- and right-handed basins and achieved high apparent coverage in the selected sampling space, yet lineage reconstruction revealed that the endpoints were reached by independent root trajectories rather than by a single continuous dynamical history. Low-dimensional collective variables can mislead by suggesting that a transition has been sampled when in reality only two independent basins have been visited. Coverage metrics identify *where* walkers have been; lineage tracking determines *whether* those

locations are dynamically connected. For any application where the central question concerns mechanism rather than mere state enumeration, this distinction is essential and cannot be inferred from projected distributions alone. The path-ensemble methods described in Chapter 5 address the complementary challenge of estimating quantitative rates once a continuous reactive pathway has been established.

The protein-folding and ligand-unbinding results demonstrate the framework's utility as an exploratory mapping tool. In folding problems dominated by entropic broadening, short adaptive walkers redistribute effort across the large and heterogeneous unfolded ensemble, increasing the probability of reaching compact native-like conformations. In ligand unbinding, the same adaptive logic surveys alternative exit directions before concentrating resources on detailed pathway or rate calculations. The two-stage WW-domain protocol, unfolding campaign followed by TICA-guided folding campaign, also illustrates a compositional usage pattern that can be adapted to other systems where the folding and unfolding mechanisms are distinct.

The MSM-seeding application makes explicit the boundary between TRAILS-MD's intended role and the domain of rigorous kinetic methods. The short adaptive trajectories are not treated as an equilibrium ensemble. Instead, they serve as a systematic mechanism for distributing initial structures across the explored landscape so that subsequent long unbiased trajectories can provide data suitable for MSM construction. The recovered CLN025 MFPTs are consistent with published values, demonstrating that this two-stage approach can yield quantitatively meaningful kinetic estimates when the follow-up production runs are sufficiently long and well distributed.

A useful point of contrast is the weighted ensemble (WE) method, which also operates by managing an ensemble of parallel trajectories. WE maintains rigorous statistical weights through a resampling procedure that splits and merges trajectories based on their progress along a predefined reaction coordinate. This procedure preserves unbiased kinetics and yields provably unbiased rate estimates, making WE the method of choice when quantitative kinetics are the primary objective. However, WE requires a suitable progress coordinate to define the binning structure, and its efficiency degrades when multiple competing pathways exist or when the relevant slow degrees of freedom are not well captured by a single coordinate. TRAILS-MD takes the opposite design philosophy: it sacrifices strict statistical rigor in exchange for flexibility and speed of exploration. By redrawing velocities at each spawn point and using heuristic density-based spawning rules, TRAILS-MD can survey diverse regions of configuration space rapidly and without requiring a predefined reaction coordinate. The resulting ensemble is not suitable for direct kinetic estimation, but it provides a practical starting point for seeding the long trajectories needed by WE, MSM, or milestoning calculations. In this sense, TRAILS-MD and WE are complementary rather than competing: the former for broad exploratory mapping, the latter for rigorous rate

determination once the relevant metastable states and pathways have been identified.

These considerations also define the principal limitations of the present implementation. The quality of any adaptive campaign depends on the chosen feature representation, spawning rule, walker length, and retraining schedule. Learned latent spaces are powerful but their axes may drift after retraining and may not always align with the physical transition of interest. Grid-based density spawning is transparent in two or three dimensions but becomes computationally impractical for higher-dimensional spaces, where geometry-based criteria (Voronoi, LOF, or FPS) are more appropriate. Both representation learning and the selection of appropriate low-dimensional coordinates are themes developed in earlier chapters of this thesis (Chapters 4 and 2). Finally, TRAILS-MD does not provide the formal rate guarantees of WE, TPS, milestoning, or related trajectory-ensemble methods. Its intended role is to complement these rigorous approaches by providing a lightweight, traceable, and easily configured platform for comparing adaptive strategies, exploring configuration space systematically, and generating well-distributed starting structures for more rigorous downstream calculations.

6.7 Summary and Outlook

This chapter presented the design, implementation, and validation of TRAILS-MD, a modular and engine-agnostic framework for lineage-aware adaptive molecular dynamics sampling. The framework integrates short parallel trajectory propagation, fixed or data-driven collective-variable spaces, interchangeable spawning policies, restartable workflow state, and explicit parent–child trajectory records within a single, self-consistent adaptive workflow. Across a hierarchy of test systems, including alanine dipeptide, CLN025, AIB9, the Trp-cage and WW domain mini-proteins, the OAMe-G2 host–guest complex, HSP90 ligand dissociation, and MSM construction from TRAILS-MD-seeded CLN025 trajectories, the framework demonstrated that the same underlying algorithm can support several related but distinct tasks: rapid conformational coverage, comparison of physical and learned sampling spaces, diagnosis of pathway connectivity through lineage analysis, exploratory mapping of ligand-exit tunnels, and systematic generation of representative starting structures for longer kinetic simulations.

Validation results established several concrete outcomes:

- For alanine dipeptide, density-based spawning achieved 60% Ramachandran coverage within 99 iterations, recovering all four conformational basins that a single unbiased trajectory of equivalent length failed to reach.
- For CLN025, the comparison of fixed physical, TVAE, and TICA sampling spaces showed that learned coordinates can resolve conformational states that are

degenerate in simple physical projections, improving both sampling quality and spawning specificity.

- For AIB9, lineage reconstruction showed that endpoint separation in supervised coordinates and configurational coverage can both be achieved without sampling a continuous dynamical transition, confirming that trajectory bookkeeping is essential for mechanistic adaptive sampling studies.
- Adaptive ensembles sampled entropic unfolded spaces efficiently to recover the native basin for Trp-cage and the WW domain and the two-stage TICA-guided protocol was an effective compositional strategy for complex folding problems.
- For HSP90, the adaptive framework was able to quickly generate a qualitative map of accessible ligand exit directions from the binding site that can be used as candidate dissociation routes for subsequent mechanistic analysis.
- For CLN025 MSM construction, a two-stage workflow combining TRAILS-MD exploration with 17.15 μs of TRAILS-MD-seeded production trajectories recovered folding and unfolding MFPTs consistent with published microsecond-scale kinetics.
- Performance benchmarking on the WW domain demonstrated 87% parallel efficiency on four GPUs, with the analysis overhead constituting only 4–7% of total iteration wall time and scaling approximately independently of the sampling-grid resolution.

The central message is that adaptive sampling should be evaluated against the specific question it is asked to answer. For well-characterized systems with known CVs, adaptive coverage in an appropriate low-dimensional space is both sufficient and rapidly achievable. For mechanistic rare-event problems, coverage must be supplemented by explicit lineage analysis. For quantitative kinetics, adaptive exploration should be treated as a preparatory stage, with rigorous kinetic estimation delegated to dedicated trajectory-ensemble methods such as WE, TPS, or milestoning.

Future developments of TRAILS-MD will focus on several directions:

1. **On-the-fly biasing integration:** Combining the exploratory adaptive walker ensemble with active bias potentials, such as metadynamics or OPES applied along the learned TVAE or Deep-TICA latent coordinates, would create a hybrid framework that accelerates both coverage and barrier crossing simultaneously.
2. **Automated hyperparameter tuning:** Algorithms that could dynamically select model lag time τ, latent dimensionality, retraining frequency, and spawning

grid resolution based on convergence diagnostics would reduce the manual configuration burden and improve robustness for new systems.

3. **Extended kinetic integration:** The direct coupling of TRAILS-MD exploration with weighted ensemble or milestoning rate calculations will allow end-to-end workflows from initial structure to quantitative kinetics without requiring separate software environments.

4. **More engine and force-field support:** Increasing the engine abstraction layer to support more MD backends and coarse-grained force fields would increase the set of molecular systems available to the framework.

By making the stages of exploration, coordinate discovery, pathway analysis, and kinetic estimation explicit, modular, and reproducible, TRAILS-MD provides a practical platform for developing and benchmarking adaptive MD workflows across molecular systems of increasing complexity and biological relevance. The software is openly available at `https://github.com/TeamSuman/Trails-MD`, with documentation, example configuration files, and benchmark systems corresponding to all applications described in this chapter.

7

Summary and Future Outlook

This dissertation examined how machine learning can strengthen molecular modeling and simulation when it is used as a tool for representation, interpretation, and sampling rather than as a replacement for molecular physics. The central problem addressed throughout the thesis is that molecular systems are high-dimensional, heterogeneous, and often governed by rare events. A simulation may contain all atomic coordinates, but those coordinates must be converted into useful molecular descriptions, physically meaningful states, and sampling strategies that can reach important regions of configuration space. The work presented here therefore follows a common thread: learn or construct representations that preserve the relevant molecular information, use those representations to identify states or pathways, and then use the resulting knowledge to guide further exploration.

7.1 General Summary

The thesis begins with the question of molecular representation in the context of aqueous solvation free-energy prediction. Chapter 3 showed that predictive accuracy on familiar data is not sufficient evidence that a model has learned transferable chemistry. Descriptor-, fingerprint-, graph-, and trajectory-derived representations all captured useful structure-property relationships, but their performance depended strongly on the chemical diversity of the training data, the physical information retained by the representation, and the downstream prediction strategy. Graph-based models that explicitly describe solute–solvent interactions gave strong internal performance, but they still failed when asked to extrapolate to size regimes and chemotypes weakly represented in the training set. Trajectory-derived embeddings and outlier-aware descriptors improved selected external predictions by adding conformational, physical, and applicability-domain information, but they did not produce a universally transferable model. This chapter therefore established transferability as a central criterion for molecular machine learning: models must be tested not only on random splits, but also across chemically distinct datasets, with their dominant failure modes and supported chemical domain reported explicitly.

Chapter 4 moved from molecular property prediction to structural identification in

condensed-phase water. The *IceCoder* framework demonstrated that local molecular environments can be mapped into a compact learned space that separates multiple ice polymorphs and liquid-like configurations. Classical order parameters remain valuable because they are interpretable and inexpensive, but they can overlap when many phases are considered under thermal fluctuations. By combining symmetry-aware local descriptors with a learned low-dimensional representation, *IceCoder* provided both local phase labels and continuous coordinates for tracking phase evolution. This result strengthened the thesis theme that learned representations are most useful when they remain connected to physical structure and can be validated against known molecular behavior.

Chapter 5 addressed rare-event sampling and pathway-resolved kinetics. *PathGennie* showed that short unbiased trajectory segments can be selected and propagated to discover transition routes much faster than brute-force simulation. These generated pathways then became the input for a more quantitative workflow: clustering, path refinement, construction of path collective variables, and Weighted Ensemble simulations. The key result is that rare events should not always be summarized by a single global rate or a single reaction coordinate. Multiple routes can connect the same initial and final states, and different routes can carry different amounts of reactive flux. The path-resolved workflow therefore links rapid pathway discovery with route-specific kinetic interpretation.

Chapter 6 generalized this sampling logic into *TRAILS-MD*, a restartable and lineage-aware adaptive sampling framework. Rather than assuming that a target pathway or final state is already known, *TRAILS-MD* iteratively runs many short simulations, projects the sampled structures into fixed or learned spaces, and selects new starting points from sparse, frontier, or outlying regions. Its defining feature is explicit trajectory lineage: every spawned segment is connected to its parent, allowing discontinuous adaptive fragments to be traced into mechanistic histories. This distinction is important because coverage of a projected space does not by itself prove that a continuous molecular transition has been sampled. By combining adaptive exploration with lineage reconstruction, *TRAILS-MD* provides a bridge between broad conformational mapping and downstream kinetic analysis.

Taken together, these chapters form a connected workflow. The first contribution asks how molecules should be represented for transferable prediction and how a model should expose the limits of its chemical applicability domain. The second asks how local molecular environments can be recognized in complex condensed phases. The third asks how rare pathways can be found and assigned kinetic meaning. The fourth asks how exploration can be automated while preserving the history needed for mechanistic interpretation. Across all four cases, the central message is the same: machine learning is most effective in molecular simulation when it improves the representation of molecular

information and remains constrained by physical validation.

7.2 Limitations and Open Questions

Several limitations remain. First, all data-driven models inherit the coverage and bias of the data used to build them. In solvation prediction, extrapolation errors arose when chemical motifs, hydrophobic size trends, shielded polar groups, ionizable neutral forms, or unsupported elements were poorly represented or not explicitly described. Trajectory-derived features reduced some of these errors, but their benefit depended on the target dataset and prediction formulation. In adaptive sampling, learned coordinates can only reflect the regions already visited by the simulation. Future workflows must therefore treat data selection, applicability-domain definition, and outlier diagnosis as part of the scientific problem, not merely as preprocessing steps.

Second, compact representations inevitably discard information. This is useful when irrelevant fluctuations are removed, but it can also hide important physical distinctions. In the ice-phase study, oxygen-centered descriptors resolve many polymorphs but cannot fully distinguish hydrogen-ordered and hydrogen-disordered pairs without additional proton-sensitive information. In rare-event sampling, a low-dimensional coordinate may merge distinct mechanisms or omit slow orthogonal motion. These examples show that dimensionality reduction must be paired with physical checks, structural interpretation, and, where possible, kinetic validation.

Third, pathway completeness remains difficult to prove. A generated set of transition paths can reveal important routes, but absence of a route in the sampled ensemble does not guarantee that the route is physically irrelevant. Similarly, adaptive sampling can improve coverage but does not automatically produce statistically rigorous rates. For this reason, exploratory methods such as *PathGennie* and *TRAILS-MD* should be viewed as mechanisms for discovery, initialization, and hypothesis generation. Quantitative thermodynamics and kinetics still require careful follow-up using methods with appropriate statistical foundations.

Finally, interpretability remains a central challenge. Learned models can separate states, predict properties, or guide spawning decisions, but a useful scientific model should also explain what molecular features control the result and when the model is being asked to extrapolate. The next generation of methods should make it easier to connect latent coordinates, trajectory-derived observables, applicability-domain flags, model uncertainty, and pathway assignments back to chemical structure, solvent organization, hydrogen bonding, conformational exposure, and other physically meaningful quantities.

7.3 Future Research Directions

7.3.1 Richer and More Transferable Molecular Representations

Future work should focus on representations that combine chemical flexibility with physical constraints. For solvation and binding problems, this means moving beyond topology alone and incorporating three-dimensional geometry, conformational exposure, solvent response, polarization, size-dependent cavity formation, and explicit applicability-domain information. The trajectory-derived representation in Chapter 3 is a step in this direction because it adds learned conformational and physical proxies to static molecular encodings, but its dataset-dependent performance shows that representation learning alone is not enough. Equivariant graph neural networks, neural interatomic potentials, and molecular foundation models offer promising routes because they can encode geometry and symmetry more naturally than fixed fingerprints or scalar descriptors [**Behler2016**, **Noe2020**, **Abramson2024**]. However, these models must be evaluated under demanding transfer tests, including cross-dataset validation, scaffold shifts, chemically targeted extrapolation, and explicit reporting of unsupported chemotypes or elements.

For condensed-phase systems, future descriptors should include the degrees of freedom most relevant to the phase distinction being studied. In ice, this means extending oxygen-centered descriptions to include proton ordering, hydrogen-bond directionality, and possibly dynamical information. More generally, local structure identification should move toward representations that are sensitive enough to distinguish subtle physical states while remaining stable under thermal noise.

7.3.2 Adaptive Sampling with Uncertainty and Feedback

The adaptive workflows developed here can be extended by making uncertainty an explicit part of the sampling decision. Instead of selecting new starting points only from sparse or frontier regions, future algorithms could combine coverage, model uncertainty, kinetic relevance, and structural diversity into a unified spawning objective. Such a strategy would allow simulations to ask not only "where have we not been?" but also "where would new data most improve the model?"

Another important direction is tighter coupling between adaptive sampling and enhanced sampling. *TRAILS-MD* and *PathGennie* could be linked with methods such as metadynamics, OPES, Weighted Ensemble, milestoning, or Markov state modeling so that exploratory trajectories directly seed rigorous thermodynamic or kinetic calculations. In this combined workflow, adaptive methods would discover useful coordinates and representative structures, while statistically grounded methods would provide final rates, populations, and uncertainties.

7.3.3 Pathway-Resolved and Mechanism-Aware Kinetics

Many molecular processes are not single-route events. Ligand dissociation, folding, nucleation, and conformational switching can proceed through competing channels whose importance changes with mutation, temperature, solvent, or ligand chemistry. Future rare-event methods should therefore retain pathway identity rather than collapsing all reactive trajectories into one global coordinate. History-labeled bins, multi-channel Weighted Ensemble schemes, non-Markovian analysis, and lineage-aware trajectory reconstruction are promising tools for this purpose.

The long-term goal is not only to estimate a rate constant, but to explain why that rate changes. A mutation may accelerate dissociation by opening one route, closing another, or redistributing flux among existing channels. A useful simulation workflow should therefore report both the total kinetic outcome and the mechanistic decomposition behind it.

7.3.4 Toward Autonomous Molecular Discovery

The broader outlook is an automated loop in which simulation, machine learning, and experimental or high-level computational data inform one another. A model could identify uncertain or unsupported regions of chemical space, request new simulations or quantum calculations, update its representation, and then guide further exploration. In materials or drug discovery settings, such a loop could be connected to automated synthesis and measurement, creating closed workflows that propose, test, and refine molecular candidates.

For this vision to be reliable, automation must not remove physical scrutiny. Models should expose uncertainty, record provenance, preserve trajectory lineage, report applicability-domain limits, and provide interpretable links between learned variables and molecular mechanisms. The most useful autonomous workflows will therefore combine algorithmic efficiency with transparent validation.

7.4 Concluding Perspective

The work in this dissertation supports a practical view of machine learning in molecular simulation. Its value lies not simply in faster prediction or more complex models, but in helping scientists decide what information matters, when a model is outside its supported domain, where simulations should go next, and how molecular events can be interpreted after they are observed. By connecting transferable prediction, local state identification, rare-event pathway discovery, and lineage-aware adaptive sampling, this thesis contributes toward simulation workflows that are more efficient, more interpretable, and better suited to the complexity of molecular systems.