8-2019

# Advancing Molecular Simulations of Crystal Nucleation: Applications to Clathrate Hydrates

Ryan Scott DeFever
*Clemson University*, rdefeve@g.clemson.edu

# Advancing molecular simulations of crystal nucleation: Applications to clathrate hydrates

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Chemical Engineering

by
Ryan S. DeFever
August 2019

Accepted by:
Dr. Sapna Sarupria, Committee Chair
Dr. David Bruce
Dr. Steven Stuart
Dr. Rachel Getman
Dr. Joseph Scott

# Abstract

Crystallization is a fundamental physical phenomenon with broad impacts in science and engineering. Nonetheless, mechanisms of crystallization in many systems remain incompletely understood. Molecular dynamics (MD) simulations are a powerful computational technique that, in principle, are well-suited to offer insights into the mechanisms of crystallization. Unfortunately, the waiting time required to observe crystal nucleation in simulated systems often falls far beyond the limits of modern MD simulations. This rare-event problem is the primary barrier to simulation studies of crystallization in complex systems. This dissertation takes a combined approach to advance simulation studies of nucleation in complex systems. First, we apply existing tools to a challenging problem — clathrate hydrate nucleation. We then use methods development, software development, and machine learning to address the specific challenges to simulation studies of crystallization posed by the rare-event problem.

Clathrate hydrate formation is an exemplar of crystallization in complex systems. Nucleation of clathrate hydrates generally occurs in systems with interfaces, and even homogeneous hydrate nucleation is inherently a multicomponent process. We address two aspects of clathrate hydrate nucleation which are not well-studied. The first aspect is the effects of interfaces on clathrate hydrate nucleation. Interfaces are common in hydrate systems, yet there are few studies probing the effects of interfaces on clathrate hydrate nucleation. We find that nucleation occurs through a homogeneous mechanism near model hydrophobic and hydrophilic surfaces. The only effect of the surfaces is through a partitioning of guest molecules which results in aggregation of guest molecules at the hydrophobic surface. The second aspect is the effect of guest solubility in water on the homogeneous nucleation mechanism. Experiments show that soluble guests act as strong promoter molecules for hydrate formation, but the molecular mechanisms of this effect are unclear. We apply forward flux sampling (FFS) and a committor analysis to identify good approximations of the reaction coordinate for homogeneous nucleation of hydrates formed from a water-soluble guest molecule. Our results suggest the possibility that the nucleation mechanism for hydrates formed from water-soluble guest molecules is different than the nucleation mechanism for hydrates formed from sparingly soluble guest molecules.

FFS studies of crystal nucleation can require hundreds of thousands of individual MD simulations. For complex systems, these simulations easily generate terabytes of intermediate data. Furthermore, each simulation must be completed, analyzed, and individually processed based upon the behavior of the system. The scale of these calculations thus quickly exceeds the practical limits of traditional scripting tools (e.g., bash). In order to apply FFS to study clathrate hydrate nucleation we developed a software package, SAFFIRE. SAFFIRE automates and manages FFS with a user-friendly interface. It is compatible with any simulation software and/or analysis codes. Since SAFFIRE is built on the Hadoop framework, it easily scales to tens or hundreds of nodes. SAFFIRE can be deployed on commodity computing clusters such as the Palmetto cluster at Clemson University or XSEDE resources.

Studying crystal nucleation in simulations generally requires selecting an order parameter for advanced sampling *a priori*. This is particularly challenging since one of the very goals of the study itself may be to elucidate the nucleation mechanism, and thus order parameters that provide a good description of the nucleation process. Furthermore, despite many strengths of FFS, it is somewhat more sensitive to the choice of order parameter than some other advanced sampling methods. To address these challenges, we develop a new method, contour forward flux sampling (cFFS), to perform FFS with multiple order parameters simultaneously. cFFS places nonlinear interfaces on-the-fly from the collective progress of the simulations, without any prior knowledge of the energy landscape or appropriate combination of order parameters. cFFS thus allows testing multiple prospective order parameters on-the-fly.

Order parameters clearly play a key role in simulation studies of crystal nucleation. However, developing new order parameters is difficult and time consuming. Using ideas from computer vision, we adapt a specific type of neural network called a PointNet to identify local structural environments (e.g., crystalline environments) in molecular simulations. Our approach requires no system-specific feature engineering and operates on the raw output of the simulations, i.e., atomic positions. We demonstrate the method on crystal structure identification in Lennard-Jones, water, and mesophase systems. The method can even predict the crystal phases of atoms near external interfaces. We demonstrate the versatility of our approach by using our method to identify surface hydrophobicity based solely upon positions and orientations of nearby water molecules. Our results suggest the approach will be broadly applicable to many types of local structure in simulations.

We address several interdependent challenges to studying crystallization in molecular simulations by combining software development, method development, and machine learning. While motivated by specific challenges identified during studies of clathrate hydrate nucleation, these contributions help extend the applicability of molecular simulations to crystal nucleation in a broad variety of systems. The next step of the development cycle is to apply these methods on complex systems to motivate further improvements. We believe that continued integration of software, methods, and machine learning will prove a fruitful framework for improving molecular simulations of crystal nucleation.

# Dedication

I dedicate this dissertation to my loving parents, Caren A. Smith and Larry F. DeFever. I cannot imagine two more supportive or caring individuals. Their positive impact on my life goes further than either of them will likely ever understand.

# Acknowledgments

First and foremost, I would like to acknowledge my research advisor, Dr. Sapna Sarupria. Her scientific curiosity, work ethic, determination, intelligence, and desire to inspire greatness in others are immediately apparent to all who know her. She forever altered the trajectory of my life when she introduced me to the fields of statistical mechanics, molecular simulations, and rare events. These intellectual pursuits captured my heart and imagination in a way that few things can. Dr. Sarupria displayed an impressive capability to know when to let me flounder, nudge me forward, or push me harder. I will miss our lively scientific disagreements. She respected and listened to my input to a degree that is rare in student–advisor interactions. Sapna also became a valued friend.

Next, I want to acknowledge my family. From the time I was a child my parents exposed me to as many opportunities as they could. Their support was absolutely unwavering in all of my endeavors. They never told me what I couldn't or shouldn't do. Growing up, my mother taught me how to construct a convincing analytical argument. My father taught me how to rearrange mathematical symbols on paper — when we weren't temporarily distracted by a strong disagreement. My mother provided the best advice on navigating life. My father eagerly listened to my latest research progress or tribulations. Somewhere along the way, I was amazingly fortunate to meet the love of my life, Corrin George. Her ideas, kindness, and patience through the nadir of my graduate studies were invaluable. She is my family now too. I hope we change the world together.

As with life, graduate school would be difficult without friends. For all the good discussions, help, and support (not to mention, patiently tolerating me) in the lab – I want to particularly acknowledge Siva Dasetty, Steven Hall, Jiarun Zhou, and Tianmu Yuan. And to the friends who would never turn down my invitation for an escape from Earle Hall – Michael Spagnuolo, Apoorva Balwani, Cameron Bodenshatz, Adam Klett, Cabell Lamie, and Allison Yaguchi.

I have crossed paths with a few individuals that had outsize influence in my life. It would be

unjust to not acknowledge their contributions to who I am today. David Dejesa, a high school history instructor who cultivated my interests in public policy, politics, and philosophy. David Dalby, a high school chemistry instructor who taught the first science class that I found intellectually stimulating. Short of this class I doubt I would have pursued a future in the sciences. Dr. Gautam Bhattacharyya, who kindly provided me with my first undergraduate research opportunity and introduced me to the intellectual journey that is research.

Last but not least, I would not have been able to complete my degree without funding and computing resources. I am grateful for support from the College of Computing, Engineering, and Applied Sciences Deans' Scholar program, Clemson University start-up funds, and the Department of Energy, Office of Science, Office of Basic Energy Sciences, Award Number DE-SC0015448. The Cyberinfrastructure Technology Integration Group is acknowledged for support and generous allocation of computing resources on the Palmetto Cluster at Clemson University.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Crystallization is the transition of matter from the liquid phase to a highly ordered solid phase. The fundamental nature of this process makes it ubiquitous in science and engineering. The canonical everyday example of crystallization is the freezing of liquid water to ice. In the natural world, crystallization is involved in phenomena from fields as diverse as geology (e.g., mineral formation) [5], biology (e.g., antifreeze proteins) [6, 7], and climate science (e.g. cloud formation) [8, 9]. In engineering applications, the objective is to manipulate or control crystallization to achieve some outcome. The chemical and pharmaceutical industries leverage crystallization as an important separation and purification technique [10]. On the other hand, preventing crystallization can also be an important engineering objective. Examples include preventing ice formation on surfaces [11, 12] or clathrate hydrate formation in oil and gas pipelines [13, 14].

Crystallization consists of two major steps: nucleation and growth. Nucleation describes the formation of the nascent crystal nucleus from the liquid, while growth considers how the boundary between liquid and crystal propagates through the liquid once a sufficiently large crystal nucleus has formed. Both nucleation and growth are inherently molecular-level processes. In experiments, the combination of small length scales and short time scales are difficult to probe. In contrast, these conditions seem precisely suited to molecular simulations. Indeed, molecular simulations can offer substantial insights into crystal growth and dissociation. Nucleation, however, remains stubbornly difficult to study in molecular simulations. Despite the macroscopic frequency of nucleation, it is an extremely rare event on the time and length scales of molecular simulation. For example, the ice nucleation rate at 235 K is an astounding $5 \times 10^{15}$ events $\mathrm{m}^{-3}\mathrm{s}^{-1}$ [15]. However, in the observation

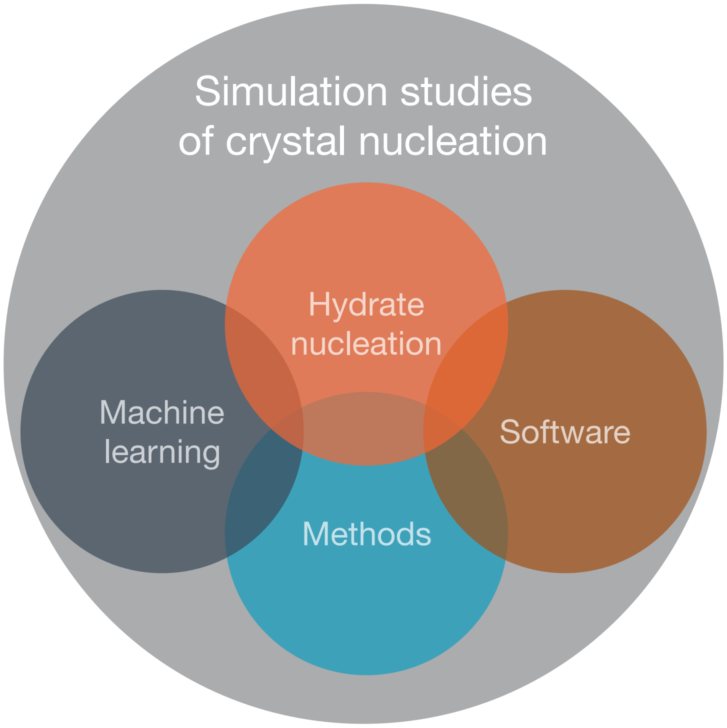

Figure 1.1: The multicomponent approach to advancing simulation studies of crystallization presented herein. This dissertation primarily builds the individual components as motivated by our experiences studying hydrate nucleation. The eventual vision imagines integrating the various components as shown by overlapping regions.

volume of a typical molecular simulation (e.g., 10×10×10 $nm^3$, or ∼33,000 water molecules) the simulation time required to observe a single nucleation event is $10^9$ s. Current computational limits are $\sim 10^{-5}$ s at best. This so-called rare event problem creates a series of challenges to studying nucleation in molecular simulations.

This dissertation encompasses an integrated approach to address the challenges of studying crystal nucleation in molecular simulations. To understand these challenges in the context of complex systems, we study clathrate hydrate nucleation. This phenomenon is well-motivated from the standpoint of engineering applications and is at the forefront of system complexity for which nucleation can currently be investigated with molecular simulations. We applied straightforward simulations and one of the best advanced sampling methods for crystal nucleation in our studies. We then proceeded to develop a substantial extension to the advanced sampling method, software, and a deep learning approach for structure identification to address specific challenges that arose in our studies of clathrate hydrate nucleation. A schematic highlighting the components of our approach is shown in Fig. 1.1. We do not believe these components represent an adequate solutions in isolation — hence motivating an integrated approach. The remainder of this introduction is devoted to the relevant background in nucleation, rare events in simulation, and clathrate hydrates.

## 1.1 Classical nucleation theory

Nucleation is the primary mechanism by which out-of-equilibrium metastable liquids relax to equilibrium [16, 17]. The quantities of interest are the rate, or frequency, at which nucleation events occur, and the molecular mechanism of nucleation. The nucleation rate describes the expected length of time that the metastable liquid will exist before it irreversibly transforms to the equilibrium crystal phase. Any good theory of crystal nucleation should thus offer predictions of the nucleation rate. We focus on the case of homogeneous nucleation, i.e., nucleation from a well-mixed homogeneous phase in the absence of any impurities or external interfaces [16]. The standard theoretical framework for describing this situation is classical nucleation theory (CNT) [18, 19, 20, 21, 22]. CNT is a classical thermodynamic theory that considers the free energy (or reversible work) required to form a nucleus of a stable phase within a bath of the metastable phase. The free energy of forming a nucleus passes through a maximum at the critical nucleus size. CNT predicts that the nucleation rate is inversely proportional to the exponential of the free energy required to form the critical nucleus. CNT assumes that the crystal nucleus has the same structure as the bulk crystal phase and that there exists a perfectly sharp interface between the crystal nucleus and liquid bath. These two assumptions comprise the capillarity approximation — that is, even infinitely small nuclei have bulk properties — and that there is an equilibrium distribution of nuclei sizes up to the critical nucleus size. The capillarity approximation is likely the source of a number of errors of CNT [16, 17].

CNT describes two contributions to the free energy of forming a nucleus with radius $R$: (1) the difference in free energy between the bulk stable and bulk metastable phases, and (2) the free energy associated with forming an interface between the stable and metastable phases. Since by definition the stable phase has a lower free energy than the metastable phase, the bulk contribution always favors the formation of larger nuclei. In contrast, the formation of an interface is energetically unfavorable; the interfacial term always disfavors the formation of larger nuclei. However, the volume and surface area of a sphere grow at different rates with increasing $R$. The difference in proportionality between the nucleus size, $R$, and the contributions of the bulk ($\propto R^3$) and interfacial

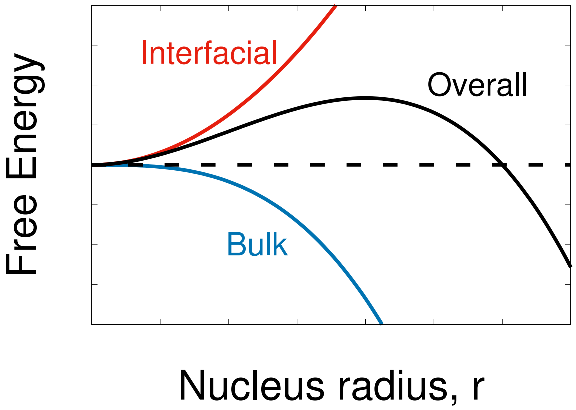

Figure 1.2: Schematic of the bulk and interfacial contributions to the free energy of forming a nucleus of a stable phase within a metastable phase. Dashed line represents the zero of free energy.

($\propto R^2$) terms to the overall free energy gives rise to an energetic profile which passes through a maximum at the critical nucleus size (Fig. 1.2). Here, growth of the nucleus becomes energetically favorable. The essence of CNT is to capture this competition between the bulk and interfacial contributions to the overall free energy of forming a nucleus. The free energy of forming of a nucleus with radius $R$ is written as:

$$\Delta G(R) = -\frac{4}{3}\pi R^3 |\Delta\mu_{\mathrm{v}}| + 4\pi R^2 \gamma \tag{1.1}$$

where $\Delta G(R)$ is the free energy of forming a nucleus with size $R$, $|\Delta\mu_{\mathrm{v}}|$ is the magnitude of the per-volume chemical potential difference between the bulk stable and bulk metastable phases, and $\gamma$ is the interfacial tension between the two phases. The critical nucleus size, $R_{\mathrm{c}}$, can be identified by setting $dG(R)/dR = 0$ and solving for $R$. The nucleation rate is written in Arrhenius form,

$$K = A \exp\left[-\frac{\Delta G(R_{\mathrm{c}})}{k_B T}\right] \tag{1.2}$$

where $K$ is the nucleation rate constant (expressed in events vol$^{-1}$ time$^{-1}$ in the homogeneous case) and $A$ is a kinetic prefactor that accounts for the density of nucleation sites, attachment rate of particles to the growing nuclei, and curvature of the free energy barrier near $R_{\mathrm{c}}$. The driving force for nucleation is the magnitude of $\Delta\mu_{\mathrm{v}}$, which equals zero at equilibrium and increases with supercooling or supersaturation. The critical nucleus size decreases with increasing driving force, approaching zero in the limit of an infinite driving force. At sufficiently large driving force, the metastable phase falls out of metastability and the transformation to the stable phase occurs via spinodal decomposition rather than nucleation [16]. Note that the rate constant often goes through a maximum before reaching spinodal decomposition. The maximum arises because the kinetic prefactor, $A$, generally decreases with increasing supercooling, while the exponential term increases with increasing supercooling [17].

Though Eqn. 1.1 only applies to homogeneous nucleation, it can be modified to describe heterogeneous nucleation (i.e., nucleation at some external surface) by including the nucleus–surface contact angle [23, 24]. Eqn. 1.1 can also be constructed as a function of the number of molecules in the nucleus, $n$, rather than nuclei radius, $R$. The idea that the nucleus size, $n$, is reaction coordinate for nucleation is inspired by CNT. This assumption is the motivation for calculating the free energy barrier or tracking the progress of nucleation as a function of $n$ when applying advanced sampling techniques to study crystal nucleation (Sec. 1.4).

Overall, CNT has been extremely successful as a descriptive theory and is considered by some [25] to be one of the great rate theories of all time. CNT is used to predict rates, trends in rates, and to interpret experimental data. Nonetheless, there are a variety of situations where the assumptions of CNT are not applicable. Non-classical nucleation mechanisms include two-step nucleation mechanisms that have been proposed for nucleation from solutions [26]. In a two-step nucleation mechanism there are two free energy barriers to nucleation. The first is ascribed to densification of solutes while the second arises from the ordering required to form a crystal. Clathrate hydrate nucleation [27], calcium carbonate nucleation from aqueous solution [28], and protein crystallization [29] are examples that may exhibit such two-step mechanisms. Even in the case of simple models of atomic fluids, evidence suggests that nucleation proceeds along both densification and ordering parameters [30, 31]. These complications highlight the complexities of crystallization.

## 1.2 Molecular simulations of crystallization

Molecular dynamics (MD) simulations are a valuable tool to supplement our understanding of crystal nucleation [17]. In general, MD simulations are useful for interrogating the molecular level behavior of systems and relating microscopic behavior to macroscopic observable properties [32, 33]. MD simulations can thus, in principle, offer molecular level insights into nucleation. In MD simulations, particle interactions are described via some interatomic or intermolecular potentials. The force felt by each particle is provided by the negative derivative of the potential. The time behavior of the system can then be generated by numerically integrating the classical equations of motion with a sufficiently small integration time step. Due to the rare-event problem described earlier in this chapter, only nucleation in the simplest systems and/or at conditions of high driving force is accessible in straightforward MD. Therefore, many of the results presented in Sec. 1.2.1 and 1.2.2 are generated from MD simulations combined with some advanced sampling method (see Sec. 1.4).

### 1.2.1 Model atomic fluids

Molecular simulations have been used to study crystallization for many years. Hard spheres are perhaps the simplest model that exhibits a fluid to solid phase transition [34]. In fact, the first molecular dynamics simulations investigated phase behavior in the hard sphere system [35]. Fortuitously, hard sphere systems have experimental analogues in the form of colloidal systems, where nucleation can be observed directly with confocal microscopy [36, 37, 38]. Simulations and experiments of hard sphere systems have revealed the following observations: (1) despite the simplicity of the systems there appears to be a large discrepancy between predicted nucleation rates from simulations and measured rates from experiments at low driving force [39, 40]. This discrepancy has persisted in spite of efforts to resolve it [41, 42]. (2) In both simulations [39, 41, 40] and experiments [38] nuclei are observed to be aspherical and have surface roughness. The former point suggests that our picture of crystal nucleation is incomplete in even the simplest systems, and the latter point is in conflict with CNT. Crystallization in the Lennard-Jones (LJ) fluid is another simple model system which has been studied extensively in molecular simulation. Results from nucleation of LJ systems are consistent with hard-sphere results in finding aspherical nuclei with rough surfaces [39, 41, 40]. LJ nuclei also exhibit varying degrees of crystallinity and polymorphism; evidence of multiple crystal phases appears in the nuclei, rather than only the most stable crystal phase [43, 44, 45]. Evidence suggests a trade-off between nucleus size and structure [44]. Somewhat smaller and more crystalline nuclei are similarly likely to be critical as somewhat larger and less crystalline nuclei. These results from very simple systems quickly cloud the description of nucleation presented by CNT. As noted in the prior section, many of these results are generated with MD in combination with advanced sampling techniques. Fortunately, by modern standards, these simple models are not too computationally costly. This enables some comparisons between the results generated with different advanced sampling methods and results from straightforward MD. Under conditions where such comparisons are possible, the observations presented above appear consistent across different methods [40, 46, 47].

### 1.2.2 Water

Moving towards increasing complexity but remaining in the realm of single-component systems, MD simulations have been used to study the liquid to solid transition in water. Understanding ice nucleation is important for fields such as cloud physics [8, 9] and food preservation [48]. MD simulations have been used to extensively study both homogeneous [49, 50, 51, 52, 53, 54] and heterogeneous ice nucleation [55, 56, 57, 58, 59, 60, 61]. Nonetheless, ice nucleation has proven challenging to study in simulations. The first straightforward MD simulation with homogeneous ice nucleation was reported in 2002 by Matsumoto *et al.* [49]. In order to shorten the waiting time for ice formation, the density of the liquid phase was lowered to 0.96 g $\mathrm{cm}^{-3}$. Assuming the accessible simulation length doubles each year, it will still be $\sim$50 years until it is possible to simulate (via straightforward MD) homogeneous ice nucleation at realistic conditions (235 K, 1 bar) with an all-atom model. Even using advanced sampling techniques, calculating the ice nucleation rate with the all-atom TIP4P/Ice model required 22 million CPU-hours [53]. Fortunately, the much higher rates of heterogeneous nucleation make it possible to use straightforward MD to simulate ice nucleation on strong ice nucleating surfaces. Studies have provided useful insights, such as how the orientations of water molecules near a surface may act as a predictor of ice nucleation [55, 60], and that small changes in surface structure can lead to large changes in surface ice nucleating ability [55]. However, the range of supercooling and surface types that can be explored remains severely limited. As such, despite substantial efforts, a comprehensive understanding of the surface properties which promote or inhibit ice nucleation is still lacking.

### 1.2.3 The seeding method

One approach to circumvent the rare-event problem in molecular simulations of crystal nucleation is seeding. Seeding [52, 62, 63, 64] widely expands the range of accessible nucleation conditions for homogeneous nucleation. Recent efforts have also extended the method to heterogeneous nucleation [65, 66]. A crystalline ice embryo (seed) is carefully equilibrated in a bath of liquid. After equilibration, a straightforward MD simulation is performed. In sufficient time the seed will either grow until the entire system becomes solid or dissociate. With this method, the seed size that has a 50% chance of growing is identified as the critical nucleus size. The free energy of forming the critical nucleus is then calculated by combining the critical nucleus size with the free energy difference between the bulk liquid and solid phases. The remaining information required to estimate the kinetic prefactor can be estimated from nuclei growth and dissociation rates near the critical size.

Seeding has been used to evaluate nucleation rates spanning nearly 200 orders of magnitude, [52] and is thus a powerful method for evaluating nucleation at a wide range of conditions. Unfortunately, seeding is far from a perfect solution. The numerical estimate of the free energy barrier is particularly sensitive to the precise definition of the nucleus [64]. Since the nucleation rate is proportional to the exponential of the free energy barrier, the estimated rate is extremely sensitive to the exact definition of the nucleus. In NaCl nucleation, modifications to the definition of the nucleus changed estimated nucleation rates by 30 orders of magnitude [64]. Beyond this problem, seeding also assumes that the crystalline seed is a perfect sphere of the bulk crystal phase. These assumptions are not always true; results from HS/LJ systems show aspherical nuclei with polymorphism, and recent theoretical calculations argue that ice nuclei are a stacking-disordered structure composed of alternating layers of the stable bulk polymorph (ice Ih), and a metastable polymorph (ice Ic) [54].

## 1.3 Clathrate hydrates

Clathrate hydrates are a solid phase that forms from water and a guest species [13, 67, 68]. Water molecules hydrogen bond to form a space-filling crystal composed of polyhedral cages, some or all of which are occupied by a guest molecule. A variety of small molecules (e.g., ethane, propane, carbon dioxide, nitrogen, hydrogen, tetrahydrofuran, etc.) can form clathrate hydrates, but the most common example is methane [68]. In general, these compounds form at low to moderate temperatures (<300 K) and elevated pressures (>0.6 MPa). Guest molecules are an integral component of clathrates. As observed in everyday experience, the stable solid phases of water in the absence of guest molecules are ices [69]. In clathrate hydrates, the guest molecules act to stabilize the polyhedral cages that they occupy through steric repulsion with water molecules. Understanding the subtle interplay between water–water, guest–water and water-mediated guest–guest interactions is thus paramount for understanding the formation of clathrate hydrates. The inherent multicomponent nature of clathrate hydrates dramatically complicates their nucleation mechanism in comparison with simple atomic fluids and pure water.

### 1.3.1 Background and applications

For over a century from the discovery of clathrate hydrates by Sir Humphrey Davy in 1811 [70], investigations of clathrate hydrates were largely driven by scientific curiosity [68]. However, since the 1930s a large quantity of clathrate hydrate research has been motivated by oil and gas industry [68]. This is because methane clathrates can form spontaneously in oil and gas flowlines and represent a severe flow assurance hazard [14]. These flowlines can contain a mixture of water and natural gas near hydrate forming conditions, and under the correct conditions, solid hydrate plugs form rapidly. Estimated annual expenditures for hydrate inhibition are over $100 million [71]. There exist thermodynamic and kinetic strategies for preventing hydrate formation [14]. The simplest of the thermodynamic approaches is to adjust temperature and pressure conditions out of the hydrate forming region of the phase diagram. This is often not feasible. More commonly, thermodynamic inhibitors (e.g., methanol, monoethylene glycol) are added to the fluid mixture in the pipeline. Large quantities of these inhibitors (e.g., ∼20–50 wt % in free water) are necessary [72, 14]. These thermodynamic inhibitors are either recovered downstream or simply considered part of the operating cost. Kinetic strategies for hydrate prevention include anti-agglomerants and kinetic hydrate inhibitors [14, 72, 73]. Anti-agglomerants are surfactants which reduce capillary adhesion between solid hydrate particles and thus act to prevent their agglomeration. In general, it is believed that kinetic hydrate inhibitors bind to the surface of growing hydrate nuclei and prevent the formation of post-critical nuclei [73]. These strategies appear promising, but the mechanism(s) of action, particularly in the case of kinetic hydrate inhibitors, remain incompletely understood.

Methane clathrate hydrates also represent an enormous potential energy resource. Natural deposits of methane clathrates exist in reserves in permafrost and beneath the sea floor [67]. Current estimates suggest more methane trapped in hydrate form than traditional natural gas resources [74, 75]. As such, there have been efforts to understand how to extract the methane from hydrate reserves [76, 77, 78, 75, 79]. Some have even proposed extracting the methane from clathrates and replacing it with carbon dioxide [75, 80]. These technologies would all benefit from a firmer understanding of hydrate formation mechanisms.

The unique molecular properties of hydrates may prove useful for technological applications. Examples include long term natural gas storage in hydrates [81, 82], hydrogen gas storage in hydrates [83, 84, 81], and gas separations [85, 86, 87, 88]. Considering the example of long-term natural gas storage, 1 mole of methane can be stored in ∼130 $\text{cm}^3$ of volume in clathrate form. At hydrate forming conditions (273 K, 20 bar) the same 1 mole of methane has a volume of ∼1070 $\text{cm}^3$ in gaseous form at the same conditions and a volume of ∼24890 $\text{cm}^3$ at standard temperature and pressure.[1] Even though the equilibrium formation pressure is 2.6 MPa at 273 K, once formed, methane hydrate pellets are remarkably stable at atmospheric pressure and ∼250 K [89]. Thus the conditions for methane storage in hydrate form are quite reasonable. Additionally recent experimental efforts have shown that tetrahydrofuran is an effective promoter of methane hydrate formation [81, 87, 82], enabling more moderate hydrate formation conditions. Unfortunately, the mechanism of this promotion remains largely unknown, making it difficult to design or predict other hydrate promoters.

### 1.3.2 Proposed nucleation mechanisms

Several mechanisms of clathrate hydrate nucleation have been proposed. The primary hypothesis include the labile cluster hypothesis [90], the local structuring hypothesis [91], the cage adsorption hypothesis [92], and the blob hypothesis [27]. The labile cluster hypothesis envisions that clusters of hydrogen-bonded water molecules form around dissolved guest molecules and then these clusters aggregate into a hydrate nucleus. The local structuring hypothesis invokes less pre-existing water structure around each dissolved guest – instead the guests first aggregate together and water structuring occurs subsequent to a sufficiently large aggregation of guest molecules. The blob hypothesis shares many similarities – amorphous ‘blobs’ of solvent separated guest molecules form as precursors to hydrate formation. An amorphous collection of polyhedral hydrate cages are born from within these regions. The cage adsorption hypothesis is based upon the observation that polyhedral hydrate cages are stabilized by the adsorption of guest molecules to the cage faces. Under this mechanism, hydrate cages that spontaneously form are stabilized by adsorbed guest molecules, and eventually form a cluster of amorphous hydrate cages that can then anneal into the crystal structure. These hypothesis all highlight the interplay of water and guest in hydrate nucleation and

[1] Hydrate volume estimated for structure I with full cage occupancy. Gas volumes estimated with the Peng-Robinson equation of state.

challenge the simple picture of crystal nucleation proposed by CNT.

### 1.3.3 Molecular simulations of hydrate nucleation

The first unbiased hydrate nucleation trajectories in molecular simulations were reported just under a decade ago [93, 94]. Since then, there have been several efforts [27, 95, 96, 97, 98, 99, 100, 101, 62, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113] to use simulations to improve our understanding of the homogeneous hydrate nucleation mechanism. Collectively, these simulation studies provide strong evidence to suggest that the initial hydrate structures are amorphous rather than crystalline solids. In most cases, the hydrate nucleus appears to emerge out of regions with locally elevated guest concentration. There remains disagreement on the exact mechanism, but some studies studies [93, 105] suggest that water order follows the formation of appropriate guest structure. It is worth noting that with few exceptions [99, 112, 111] these studies focused on sparingly soluble guest molecules (e.g., $CH_4$ and $CO_2$). It is unknown if the nucleation mechanism changes with guest solubility. There have also been relatively fewer studies of heterogeneous nucleation [114, 115, 102, 116, 117, 118] despite the prevalence of interfaces in hydrate-forming systems. Our studies of hydrate nucleation attempt to address those two shortcomings in the current understanding of hydrate nucleation.

## 1.4 Advanced sampling methods

The prior sections demonstrated the potential of molecular simulations to provide insights into crystal nucleation and the challenges associated with studying nucleation in simulations, namely, the rare-event problem. Next, we discuss advanced sampling techniques that can help overcome the rare-event problem and supplement straightforward MD simulations. The rare-event problem is not limited to molecular simulations of crystal nucleation. It appears in a variety of other contexts, from complex processes such as protein folding [119] to apparently simpler processes such as ion-pair dissociation [120, 121]. Imagine two well-defined stable states, $A$ and $B$. The core feature of a rare event is that the waiting time ($\tau_{\text{stable}}$) between $A \to B$ or $B \to A$ transition events is orders-of-magnitude longer than the duration of the transition itself ($\tau_{\text{trans}}$). That is, $\tau_{\text{trans}} \ll \tau_{\text{stable}}$. If the object of our research is a physical process that exhibits this behavior, we are interested in (1) the mechanism by which the transition occurs and (2) the transition rate constant. The rate constant

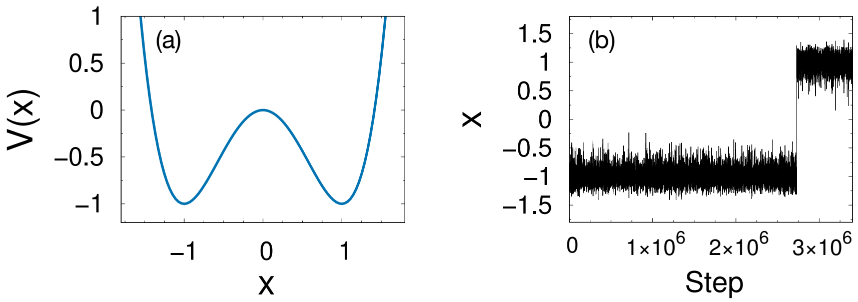

Figure 1.3: An example rare event. (a) 1D double well potential energy surface. (b) Position of a particle during a simulation of a particle on the potential from (a).

for an $A \to B$ transition can be defined as the transition probability per unit time, inverse mean residence time in $A$, or inverse mean first passage time to $B$ [122]. These three definitions become equivalent for any process where $A$ and $B$ are separated by a sufficiently large free energy barrier [123].

A schematic of a simple example of a 1D double-well potential is shown in Fig. 1.3(a). The system has two stable states, one at $x < 0$ (state $A$) and one at $x > 0$ (state $B$). If we simulate a particle in this potential with Langevin dynamics at appropriate conditions the A→B transition displays the primary attributes of a rare event. The system samples state A for >2,000,000 steps before quickly (∼1000 steps) transitioning to state B. This demonstrates the separation of time scales between $\tau_{\text{trans}}$ and $\tau_{\text{stable}}$. The challenge is thus to accurately simulate the process (i.e., explicitly account for the shorter-timescale physics governing the transition) but run a simulation long enough to observe the transition. A variety of methods have been developed to address this problem.

Umbrella sampling [124], metadynamics [125, 126], aimless shooting transition path sampling (TPS) [127, 128], and forward flux sampling [129, 1] (FFS) are the primary advanced sampling methods which have been applied to study crystal nucleation [40, 50, 130, 51, 53, 58, 131, 132, 133, 134, 54, 61, 46, 47]. Umbrella sampling and metadynamics are free energy methods which can be used to estimate the free energy barrier to nucleation. These methods require a good order parameter, i.e., one which closely approximates the true reaction coordinate [135], to (1) calculate a physically meaningful free energy barrier and eliminate hysteresis [136] , and (2) ensure an efficient rate calculation. Unfortunately, good order parameters are difficult to know *a priori.* Furthermore, most of the free energy methods do not provide estimates of the nucleation rate nor dynamical nucleation trajectories without additional simulations. One classic approach to calculate the rate constant from free energy methods is a two-step procedure [137, 138, 123] based upon dynamically corrected [139] transition state theory [140, 141]. The first step is to calculate the free energy profile along some order parameter, and the second step is to release trajectories from the dividing surface (i.e., the maximum in the free energy profile) to estimate the transmission coefficient. The method is exact, but becomes detrimentally inefficient in the case of a poorly chosen order parameter [136, 123, 142].

FFS [129, 1] and TPS [143, 120, 127, 128] are path sampling methods which generate an ensemble of transition paths that can be used to investigate the transition mechanism and search for the reaction coordinate [144, 127, 128] of crystal nucleation [54, 133, 134, 131]. The first path sampling method was TPS [143, 120, 145]. Starting with an initial path connecting the two stable states, TPS performs a Monte Carlo random walk in path space to generate an entire ensemble of paths connecting these two states [145]. Imagine the ensemble of all possible paths of some length ($\tau_{\text{trans}} < \tau < \tau_{\text{stable}}$) that start in $A$ in the example from Fig. 1.3. An overwhelming majority of paths would be $A \to A$. The $A \to B$ paths would represent a tiny subensemble of all paths. TPS provides an efficient means for sampling the $A \to B$ subensemble. A particular benefit of TPS is that only the bounds of the two stable states ($A$ and $B$) must be defined. Aimless shooting TPS combined with maximum likelihood [127, 128] is an excellent tool for evaluating nucleation mechanisms and identifying good order parameters, but it does not provide the nucleation rate. The challenges associated with calculating the transition rate constant using TPS inspired the development of transition interface sampling (TIS) [146], and FFS [129, 1]. Relevant to our work, the TPS rate constant calculation is particularly difficult in the case of diffusive processes (e.g., nucleation) [146, 123].

### 1.4.1 Transition interface sampling and the effective positive flux

TIS introduced a simplified procedure for calculating the $A \to B$ rate constant. This depends on defining history-dependent regions of phase space. Let us define phase space as $\{x\} \subset \mathbb{R}^{6N}$, where $N$ is the number of particles, and there are $3N$ position coordinates and $3N$ momenta coordinates. Previously, initial state $A$ and final state $B$ were defined with characteristic functions as follows: for some phase point $x$, $h_A(x) = 1$ if $x \in A$ and 0 otherwise. Similarly, $h_B(x) = 1$ if $x \in B$ and 0 otherwise. TIS defined *overall* states $\mathcal{A}$ and $\mathcal{B}$. These states not only account for the current state of the system, $x$, but also the path *history.* Overall state $\mathcal{A}$ comprises all phase points in $A$ *and* all phase points which belong to trajectories that were *more recently* in $A$ than $B$. I.e., upon tracing the path backwards from $x$ one would reach $A$ before $B$. This means excursions from $A$ that have not yet reached $B$ nor returned from $A$ still belong to overall state $\mathcal{A}$. Overall state $\mathcal{B}$ is defined likewise. The rate constant can then be written as

$$k_{AB} = \frac{\langle h_{\mathcal{A}}(x_0)\dot{h}_{\mathcal{B}}(x_0)\rangle}{\langle h_{\mathcal{A}}(x_0)\rangle} \tag{1.3}$$

where the overdot indicates a time derivative, $\langle ... \rangle$ indicates an ensemble average, and $x_0$ represents the initial phase point of some path. The denominator counts the fraction of points in phase space which belong to trajectories more recently in $A$ than $B$. The numerator tabulates the fraction of those points for which $h_{\mathcal{B}}$ is changing from 0 to 1, i.e., phase points which are entering state $B$ from overall state $\mathcal{A}$. This definition aligns with our prior definition of a rate constant, the probability of transitioning to $B$ per unit time in $A$. In the limiting case, where $\tau_{\text{trans}}/\tau_{\text{stable}} \to 0$, the two definitions become identical. For any rare event the rate constant is effectively impossible to calculate directly from Eqn. 1.3 as the value of the numerator is vanishingly small. Thus, TIS introduced a factorization by defining a series of phase space regions between $A$ and $B$. The complete derivation can be found elsewhere [146]. First let us define the stable states A and B as $\{x|\lambda(x) < \lambda_A\}$ and $\{x|\lambda(x) > \lambda_B = \lambda_N\}$, respectively, where $\lambda$ is some order parameter that is a function of phase space. In practice, $\lambda$ is often defined in configuration space rather than the complete phase space. Then the phase space regions, $\Omega^{[0]+}$, $\Omega^{[1]+}$, ..., $\Omega^{[N]+}$, are defined as follows: $x \in \Omega^{[i]+}$ if $\{x|\lambda(x) > \lambda_i \wedge h^b_{A,\lambda_i}(x) = 1\}$, where $h^b_{A,\lambda_i}(x)$ is 1 if, upon following the trajectory backwards in time from $x$, we reach state A before $\lambda(x) > \lambda_i$, and 0 otherwise. Eqn. 1.3 can be recast as $k_{AB} = \langle h_{\Omega^{[N]+}}(x_0)\rangle / \langle h_{\mathcal{A}}(x_0)\rangle$, or the fraction of total phase points in overall state $\mathcal{A}$ which belong to the $\Omega^{[N]+}$ ensemble. Once again, $h_{\Omega^{[i]+}}(x)$ is a characteristic function with a value of 1 if $x \in \Omega^{[i]+}$ and 0 otherwise. The rate expression can then be written in the apparently simple form:

$$k_{AB} = \Phi_{A0} \prod_{i=1}^{N-1} \mathcal{P}(\lambda_{i+1}|\lambda_i) \tag{1.4}$$

where $\mathcal{P}(\lambda_{i+1}|\lambda_i) = \langle h_{\Omega^{[i+1]+}}(x_0)\rangle / \langle h_{\Omega^{[i]+}}(x_0)\rangle$ and $\Phi_{A0} = \langle h_{[0]+}(x)\rangle / \langle h_{\mathcal{A}}(x)\rangle$. The flux, $\Phi_{A0}$, can be evaluated directly from a long straightforward simulation in state $A$ by counting the number of first crossings of $\lambda_0$, $N^{\lambda_0}_{\text{cross}}$. By *first crossing*, we mean that crossings of $\lambda_0$ are only counted the trajectory must have more recently been in $A$ than $\lambda > \lambda_0$. Then, $\Phi_{A0} = N^{\lambda_0}_{\text{cross}}/t_{\mathcal{A}}$, where $t_{\mathcal{A}}$ is the total length time the simulation spends in *overall* state $\mathcal{A}$. For the latter part of Eqn. 1.4, consider the ensemble of all paths that begin in $A$, end in $A$ or $B$, and have at least a single crossing of $\lambda_i$. $\mathcal{P}(\lambda_{i+1}|\lambda_i)$ is equal to the fraction of those paths which also cross $\lambda_{i+1}$ at least once. TIS thus reduces the rate calculation to performing importance sampling in path space to generate a path ensemble associated with each interface, $\lambda_i$. Presuming sufficient overlap between adjacent path ensembles, the factorization in Eqn. 1.4 provides an accurate estimate of $k_{AB}$. Details of the algorithm can be found elsewhere [146]. Suffice it to say that the algorithm is non-trivial to implement and parallelize in high-performance computing environments. In particular, molecular simulations of complex systems generally require highly optimized softwares that include CPU and GPU parallelization. As such, the source code modifications required to implement many advanced sampling methods are becoming more and more difficult for researchers. It is worth noting that two software packages [147, 148, 149] have very recently been released for TIS, which should help increase the accessibility of the method.

### 1.4.2 Forward flux sampling

The primary reason for introducing the above formalism is to explain FFS. FFS extends TIS to non-equilibrium systems. Whereas TIS requires time reversible dynamics and an *a priori* knowledge of the equilibrium phase space distribution [146], FFS does not require either. Perhaps more importantly as a practical matter, the FFS algorithm remains applicable to equilibrium systems and is straightforward to understand, implement, and parallelize. FFS was originally developed to study bistable biochemical switches in gene regulatory networks [129] and shares similarities with a method [150, 151, 152] from telecommunications modeling [153]. However, it has since been applied to biomolecular conformational changes [154, 155, 156], and homogeneous [40, 50, 51, 104, 53, 108, 46] and heterogeneous [157, 58, 61] crystal nucleation. FFS uses the same effective positive flux expression as TIS to express the rate constant. However, the method of path generation is different. Whereas TIS is based upon the shooting move and integrates paths forwards and backwards in time from the shooting point, FFS uses a splitting move [158, 159] and only integrates the dynamics forwards in time from the splitting point [129, 1]. FFS thus requires stochasticity to ensure path divergence following the splitting move, although some have suggested that the Lyapunov instability may provide sufficient stochasticity in otherwise deterministic systems [153].

There are several variants of FFS, described in complete detail elsewhere [1, 153]. The most common, ‘direct FFS’ algorithm, is described below: (1) Select an order parameter, $\lambda$, that can differentiate between stable states $A$ and $B$. (2) Run a long straightforward simulation in $\mathcal{A}$. The purpose of this simulation is to decide the locations of $\lambda_A$ and $\lambda_0$, calculate $\Phi_{A0}$, and collect a large number of first-crossing phase points at $\lambda_0$ ($\sim 10^2$–$10^4$). (3) Initiate a total of $M$ simulations ($\sim 10^3$–$10^5$) from the phase points at $\lambda_0$. Velocity perturbation at the start of the simulations or stochasticity in the dynamics will assure trajectory divergence. Each simulation is continued until it reaches $\lambda_1$ or returns to $A$. Save phase points at $\lambda_1$ and discard the phase points that returned to $A$. Determine the number of simulations which reached $\lambda_1$, $N^{\lambda_1}$. Calculate the probability, $\mathcal{P}(\lambda_1|\lambda_0) = N^{\lambda_1}/M$. (4) Replace $\lambda_0$ with $\lambda_i$ and $\lambda_1$ with $\lambda_{i+1}$ and iterate on step (3) until $\mathcal{P}(\lambda_{i+1}|\lambda_i) = 1.0$ or state B is reached. The rate constant can then be calculated from Eqn. 1.4. Complete transition paths can be generated by connecting the partial paths backwards from $B$ to $A$.

Despite the apparent simplicity of the algorithm, note that FFS satisfies the requirements of the expression presented in Eqn. 1.4. The initial points collected at $\lambda_0$ are only first-crossing phase points, i.e., $h_{A,\lambda_0}(x) = 1$ for all the phase points collected at $\lambda_0$. Simulations initiated from the phase points at $\lambda_0$ ($\lambda_i$) are terminated when they reach $\lambda_1$ ($\lambda_{i+1}$). Therefore, all points collected at $\lambda_1$ ($\lambda_{i+1}$) meet the criteria of $h_{A,\lambda_1}(x) = 1$ ($h_{A,\lambda_{i+1}}(x) = 1$). Thus, the fraction of trajectories that reach $\lambda_{i+1}$ from $\lambda_i$ in FFS accounts for the complex conditional probability encoded in Eqn. 1.4. There have been efforts to develop methods to optimize the placement of $\lambda_A$ and $\lambda_0$ [155, 156], optimize the intermediate interface placement [160], and select the locations of the interfaces $\lambda_1$, $\lambda_2$, ..., $\lambda_N$ on the fly [161]. The details of our implementation of FFS is provided in later chapters.

The algorithm presented in the prior paragraphs highlight the relative simplicity and ease of parallelization of FFS. Each interface is run sequentially, and all $M$ simulations for an interface can be run in parallel, rendering the algorithm embarrassingly parallel. Source code modifications to the simulation software are not necessary, as simulations can be run for a set period of time with trajectories written to disk and then analyzed with a separate code. If the simulation crosses $\lambda_{i+1}$ the phase point at which it crossed is extracted from the saved trajectory. If the simulation fails to reach $\lambda_{i+1}$ or $\lambda_A$ in the set simulation time, the simulation is extended as required until it reaches one of the terminating conditions. These aspects of the method are especially appealing when viewed in light of the other strengths of FFS (direct evaluation of the nucleation rate, access to dynamical transition paths). All these features combined to make FFS compelling and our initial method of choice. Nonetheless, like most methods, FFS has downsides. Because of the use of the splitting move, many paths in FFS can originate from a single point at $\lambda_0$ [123, 131]. This can result in correlated transition paths. We encountered this challenge in the FFS calculations presented in Chapter 3. Another challenge is that FFS is somewhat more sensitive to the choice of order parameter than other methods (e.g., TIS). While the results of FFS are insensitive to the choice of order parameter in the limit of infinite sampling, both the efficiency and accuracy of FFS can be affected by the choice of order parameter in realistic situations [136, 162, 123]. Despite these challenges, we believe FFS represents one of the best platforms for studying crystal nucleation in molecular simulations.

## 1.5 Summary

The goal of this dissertation is to improve our ability to study crystal nucleation in molecular simulations and increase our understanding of the nucleation mechanism(s) of clathrate hydrates. We start by applying molecular simulations to study the heterogeneous and homogeneous nucleation of clathrate hydrates and then proceed to address the challenges identified in these studies in the following chapters. Since surfaces are ubiquitous in systems with hydrate nucleation, Chapter 2 explores the effects of model hydrophobic and hydrophilic surfaces on hydrate nucleation. We find two surprising results – the primary surface effect is to alter the bulk guest concentration, and regions of high guest concentration near the hydrophobic surface did not promote hydrate nucleation. Following our findings in Chapter 2 and given that homogeneous hydrate nucleation remains insufficiently well-understood, we apply FFS to rigorously study the mechanism of homogeneous hydrate nucleation in Chapter 3. We find that the hydrate nucleation mechanism appears to differ for hydrates formed from soluble guest molecules (e.g., tetrahydrofuran) compared with sparingly soluble guests (e.g., methane). Chapters 4 and 5 are dedicated to addressing some of the challenges of FFS. Chapter 4 presents software developed in collaboration with members of the School of Computing at Clemson University to deploy FFS at large scale in high-performance computing environments. This software enabled the FFS calculations that were performed in Chapter 3. In Chapter 5, we develop a method (contour forward flux sampling) that allows FFS to be performed along multiple order parameters simultaneously. The purpose of the method is help address the FFS dependence on the choice of order parameter by enabling multiple order parameters to be used simultaneously. Since good order parameters are extremely valuable for studies of nucleation, in Chapter 6 we develop a method that uses deep learning to distinguish between different crystal phases. The method is generalizable and should provide a route to rapidly develop order parameters for crystal nucleation in novel systems. Future research directions and closing remarks are provided in Chapter 7.

# Chapter 2

# Surface chemistry effects on heterogeneous clathrate hydrate nucleation[1]

## 2.1 Introduction

Interfaces are ubiquitous in applications of clathrate hydrates. Hydrate plugs form in gas pipelines where there are several examples of fluid–fluid and fluid–solid interfaces [164]. Naturally occurring gas hydrates form in arctic and oceanic sediments [77]. In fact, calculations suggest that all methane hydrate nucleation must be heterogeneous because the estimated homogeneous nucleation rate is extraordinarily low. One calculation even suggests that within a volume of water equivalent to earth's oceans one would have to wait orders of magnitude longer than the history of the known universe to observe a single homogeneous nucleation event [62]! Manipulation of surface properties therefore provides a route to control hydrate nucleation for various technological applications.

Clathrate hydrate nucleation is difficult to study in experiments due to the inherently small length and short time scales of the process [165, 166]. Meanwhile, hydrate nucleation is a rare event on time scales easily accessible to molecular simulations. Accordingly, studies of clathrate hydrate nucleation in molecular simulations are a relatively recent development. The first unbiased

[1] Material for this chapter adapted from Ref. [163]

homogeneous methane hydrate nucleation trajectories were reported under a decade ago [94], and a plethora of simulation studies of homogeneous nucleation have since followed [27, 101, 104, 108]. We direct the reader to recent reviews by English and MacElroy [167] and Barnes and Sum [168] for a more complete summary of such studies. Despite significant progress in understanding the mechanism of clathrate hydrate nucleation, many fundamental questions remain unanswered.

One such open question is how interfaces affect hydrate nucleation [166, 17]. Thus far, most studies of heterogeneous hydrate nucleation have investigated the nucleation of methane or carbon dioxide ($CO_2$) hydrates near hydroxylated silica surfaces [114, 115, 117, 116] or ice [102]. In simulations of $CO_2$–water solution in contact with completely hydroxylated silica surfaces, an ice-like layer first formed on the silica. An intermediate layer formed on the ice-like layer, and hydrate nucleated on this intermediate layer [114]. Another study of similar systems found that as the hydrophilicity of the silica surface was reduced by replacing hydroxyl groups with hydrogen atoms, no ice-like layer formed. Instead, hydrate nucleated directly on an intermediate water layer which formed on the silica surface [117]. In 3-phase silica-$CO_2$-water systems, hydrate nucleation occurred at the silica surface near the 3-phase contact line [115]. In this case, no ice-like layer was reported and hydrate cages were hydrogen-bonded to the surface either directly or through one additional water molecule.

A study of methane hydrate nucleation on silica surfaces found that nucleation occurred near the hydroxylated silica surface and that half-cage structures formed on the surfaces to reduce the mismatch between the surface and hydrate crystal structures [116]. Meanwhile, a study of hydrate nucleation in front of a growing ice front found that methane accumulated at the ice–solution interface, inducing hydrate-like defects in the ice structure which eventually resulted in hydrate nucleation slightly further from the ice–solution interface [102]. Once hydrate nucleation occurred, a liquid-like layer separated the ice and hydrate crystals.

These studies provide the first insights into the effects of surfaces on clathrate hydrate nucleation. Despite similarities in the systems studied, a diversity of mechanisms are observed, highlighting the complexity of the problem. Compared with heterogeneous ice nucleation, heterogeneous hydrate nucleation is more involved, in part due to the fact that clathrate hydrates are multicomponent. The surface can thus play multiple roles, from reducing the crystal nucleus–liquid surface area as in traditional heterogeneous crystal nucleation [23, 169], to affecting the local water structure, local guest concentration, or water-mediated solute–solute interactions in the vicinity of the surface.

To begin elucidating the various factors that may affect heterogeneous hydrate nucleation, we begin with a fundamental question: How do model hydrophilic and hydrophobic surfaces affect hydrate nucleation? -OH and -$CH_3$ terminated self-assembled monolayers (SAMs) are selected as model hydrophilic and hydrophobic surfaces, respectively. SAMs are well-characterized in both experiments and simulations, and their effects on interfacial water have been extensively studied [170, 171, 172, 173]. Furthermore, though SAMs are semi-crystalline structures, the terminal groups have some mobility, reducing the effects of lattice matching (or lack thereof) on our results.

We perform molecular dynamics (MD) simulations of clathrate hydrate nucleation from a water–guest solution in contact with SAMs to study the effect of surface hydrophobicity on hydrate nucleation. We do not observe heterogeneous nucleation on either hydrophilic or hydrophobic SAM surfaces. Notably, nucleation occurs homogeneously in the bulk solution (>1.0 nm from the surface). The primary surface effect on nucleation arises from its influence on the bulk guest concentration. Interestingly, $CH_3$SAM hinders nucleation despite significantly increased guest concentration near the surface. We surmise that the $CH_3$SAM surface prevents hydrate nucleation in the interfacial region because guest–guest contact pairs are formed more easily near this surface.

## 2.2 Computational Methods

A coarse-grained model is employed for the SAM surfaces to be compatible with the coarse-grained monatomic water model (mW) [174]. mW is computationally efficient and faithfully reproduces many properties of water [175]. Furthermore, mW has been extensively used to study ice and hydrate nucleation [27, 95, 104, 105, 51, 108, 176, 177, 178, 179, 180, 181, 182]. XL guest [95], a water-soluble guest molecule that occupies only the large cages of sII hydrate, is used. Though XL has no directional (i.e. hydrogen-bonding) interactions with mW, it can be thought of as a loose analogue of THF, in that THF is miscible with water, forms sII hydrate, and only occupies the large $5^{12}6^4$ cages of the sII crystal. Since XL is a water-soluble guest molecule, our simulations do not require high supersaturation or a multiphase system as commonly employed for studies of hydrate nucleation of sparingly soluble guest molecules. The melting point of the sII crystal with XL guest and all cages occupied was identified as 312 K through the direct coexistence method.

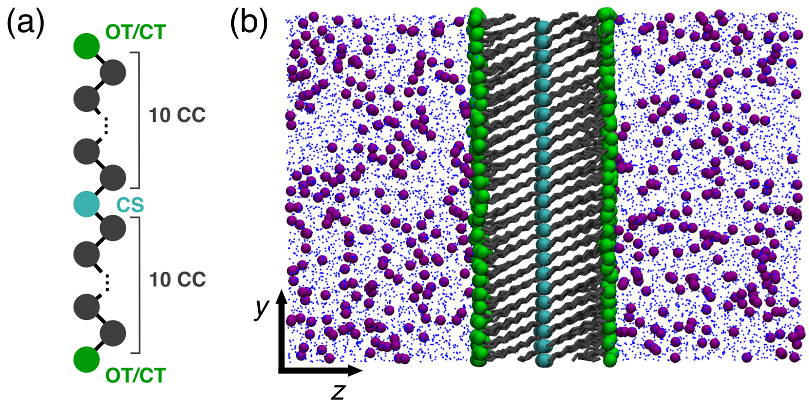

Figure 2.1: (a) Schematic of two alkane chains extending from a single central sulfur atom that comprise our SAM surfaces. (b) System setup for nucleation simulations. Alkane chain carbon atoms (CC) are shown as gray bonds, central sulfur (CS) atoms are shown as cyan spheres, -OH/-$CH_3$ terminal groups (OT/CT) are shown as green spheres, XL guest molecules are shown as purple spheres, and mW water is shown as blue points. The simulation cell is approximately $6.9 \times 6.0 \times 8.8$ $nm^3$.

### 2.2.1 Generation of SAM surfaces

The SAM surfaces were modeled as a bilayer consisting of two monolayers extending in opposite directions from a layer of central sulfur atoms. Two 10 carbon alkane chains terminated by an -OH or -$CH_3$ terminal group were covalently attached to each central sulfur atom (see Figure 2.1(a)). 192 central sulfur atoms were placed in a hexagonal close packed arrangement in the $x$-$y$ plane and spaced 0.497 nm apart to mimic the adsorption of sulfur groups on an Au (111) surface [170]. The resulting SAM bilayers were periodic in the $x$ and $y$ directions. Each monolayer had a surface area of $6.9 \times 6.0$ $nm^2$. The SAM surfaces were energy minimized and then equilibrated for 10 ns in the NVT ensemble at 300 K to allow the SAM chains to relax to their equilibrium tilt angle. An example of the resulting surface can be seen in Figure 2.1(b).

### 2.2.2 Force field details

The coarse grained SAM surfaces consist of three atom types: central sulfur (CS), alkane chain carbon (CC) and a -OH (OT) or -$CH_3$ (CT) terminal group (see Figure 2.1(a)). OHSAM and $CH_3$SAM contain CS and CC atom types, but are differentiated by the presence of either OT or CT, respectively. The CS–CS, CS–CC, CC–CC, CS–CT/OT, and CC–CT/OT non-bonded interactions were described with a Lennard-Jones (LJ) potential using parameters from the OPLS-UA force field [183, 184]. The CT–CT and OT–OT terminal group non-bonded interactions were described via the functional form of the Stillinger-Weber (SW) potential [185] as shown in Eqs. 2.1–2.3:

$$E = \sum_i \sum_{j>i} \phi_2(r_{ij}) + \sum_i \sum_{j\neq i} \sum_{k>j} \phi_3(r_{ij}, r_{ik}, \theta_{ijk}) \tag{2.1}$$

$$\phi_2(r_{ij}) = A\epsilon \left[ B \left( \frac{\sigma}{r_{ij}} \right)^p - 1 \right] \exp\left( \frac{\sigma}{r_{ij} - a\sigma} \right) \tag{2.2}$$

$$\phi_3(r_{ij}, r_{ik}, \theta_{ijk}) = \lambda\epsilon[\cos\theta_{ijk} - \cos\theta_0]^2 \exp\left( \frac{\gamma\sigma}{r_{ij} - a\sigma} \right) \exp\left( \frac{\gamma\sigma}{r_{ik} - a\sigma} \right) \tag{2.3}$$

where $E$ is the total potential energy of the system, $\phi_2$ is a two-body term and $\phi_3$ is a three-body term that allows a tetrahedral geometry to be enforced without explicit hydrogen-bonding. $r_{ij}$ is the distance between atoms $i$ and $j$, $\sigma$ is related to the atomic size, and $\epsilon$ scales the interaction strength. $\theta_{ijk}$ is the angle formed between atoms $i$, $j$, and $k$. $A = 7.049556277$, $B = 0.6022245584$, $a = 1.8$, and $\gamma = 1.2$ are constants. $\lambda = 23.15$ for mW and $\lambda = 0$ for all other three-body interactions. The CT terminal groups were type M [95]. The OT terminal groups were described as mW [174]. All bonded interactions were taken from the OPLS-AA force field [186]. Additionally, each CS atom was bonded to its six nearest CS neighbors with an equilibrium bond length of 0.497 nm and spring constant of 300.0 kcal/mol to maintain their relative positions and emulate adsorption of the SAM chains to a solid substrate.

Water and guest molecules were described by the coarse-grained mW [174] and XL guest [95] models, respectively. All interactions involving water or guest molecules were described with the functional form of the SW potential. The mW–mW, XL–XL, mW–XL, OT–mW, CT–mW, and OT–XL interactions were used exactly as they have been parameterized elsewhere [174, 95]. The CT–XL interaction parameters were calculated with Lorentz-Berthelot mixing rules between types M and XL, yielding $\epsilon_{\mathrm{CT-XL}} = 0.340$ kcal/mol and $\sigma_{\mathrm{CT-XL}} = 0.429$ nm.

The CC–mW and CC–XL interactions required further parameterization. The standard potentials for mW and XL are softer than the traditional LJ potential as the repulsive term scales by $(1/r_{ij})^4$ ($p = 4$ in Eq. 2.2) compared with $(1/r_{ij})^{12}$ for the LJ potential. This results in excessive penetration of mW and XL molecules into the SAM surfaces. Such penetration of water or guest molecules was not observed in all-atom simulations. Therefore, the CC–mW and CC-XL non-bonded interaction parameters were modified to be more repulsive by changing the exponent on the repulsive term from $p = 4$ to $p = 12$. For CC-mW interactions, $\epsilon$ and $\sigma$ values were selected to match the distance to, and depth of, the potential minimum with the potential minimum of the original M–mW interactions. The CC–XL interactions were weakened with values of $\epsilon_{\mathrm{CC-XL}} = 0.1$ kcal/mol and $\sigma_{\mathrm{CC-XL}} = 0.45$ nm. Note that the interactions between the terminal groups (CT/OT), and mW or XL remain unaffected by these modifications.

### 2.2.3 Contact angle calculations

The water contact angles on OHSAM and $\mathrm{CH_3}$SAM surfaces were calculated to quantify their hydrophobicity. The contact angle was calculated for droplet sizes of 1000, 2000, 4000, and 8000 water molecules. The larger droplets required SAM surfaces up to $15 \times 15$ nm$^2$. The profile of the droplet was determined by identifying edge of the droplet as the point where the water density fell below 1/2 of bulk water density. The interior angle between the tangent line to the droplet at the surface and the plane of the surface determined the contact angle. The macroscopic contact angle is calculated by extrapolating a linear fit of $1/r$ vs. $\theta_c$ to $1/r = 0$, where $r$ is the droplet radius and $\theta_c$ is the microscopic contact angle [187]. We find that a droplet of water on OHSAM is completely wetting (i.e. contact angle of $\sim 0°$). The contact angle of water on $\mathrm{CH_3}$SAM is 86°. These calculations confirm that the OHSAM and $\mathrm{CH_3}$SAM act as hydrophilic and hydrophobic surfaces, respectively.

### 2.2.4 Nucleation simulations

MD simulations of clathrate hydrate nucleation were performed for systems with a mW–XL solution in contact with either OHSAM or $\mathrm{CH_3}$SAM in the $NpT$ ensemble at 5.6 mol% XL, $p = 1$ atm, and $T = 230$ K and $T = 233$ K. 3.0 nm slabs of mW–XL solution, with randomly generated coordinates for mW and XL molecules, were placed on each side of equilibrated SAM surfaces. A snapshot of the SAM–solution system setup is shown in Figure 2.1(b). The SAM–solution configurations were energy minimized and equilibrated for 5 ns at $p = 1$ atm and $T = 300$ K, and then simulated in production for an additional 25 ns at the same conditions. The coordinates were stored every 1 ns from 16–25 ns to provide up to 10 independent initial configurations for the nucleation simulations. Random velocities were drawn from a Gaussian distribution to start the nucleation simulations at $T = 230$ K or 233 K. The nucleation simulations were performed until nucleation occurred or for 1 $\mu$s.

Simulations of homogeneous (bulk) nucleation of an XL–mW solution were performed in the $NpT$ ensemble at $p = 1$ atm, 4.0 mol% and 6.5 mol% XL, and $T = 230$ K and $T = 233$ K. The initial configurations for the production nucleation simulations were generated in the same manner as above; 5 ns equilibration at $p = 1$ atm and $T = 300$ K, and a 25 ns production simulation at $p = 1$ atm and $T = 300$ K with configurations written every 1 ns from 16–20 ns to generate five unique starting configurations for the production nucleation simulations.

### 2.2.5 Simulation details

All MD simulations were performed in LAMMPS (May 15, 2015 build) [188, 189] with a time step of 5 fs. Periodic boundary conditions were employed in the $x, y$, and $z$ directions. Non-bonded interactions were cutoff at 1.0 nm. All equilibration simulations were performed with the Berendsen thermostat and barostat with time constants of 0.5 ps and 1.0 ps, respectively. Production simulations were performed with the Nosé–Hoover thermostat and barostat with time constants of 1.0 ps and 5.0 ps, respectively. The SAM and solution were thermostated separately and pressure coupling was anisotropic. All snapshots were generated with Visual Molecular Dynamics [190].

## 2.3 Results and Discussion

### 2.3.1 Hydrate nucleation time

Since nucleation is a stochastic process we gathered five independent nucleation trajectories of OHSAM and $CH_3$SAM systems at 230 K. The induction time and nucleation time for each trajectory were determined by fitting a stretched exponential to the potential energy (PE) trace. Details of this procedure can be found in the supplemental material of Ref. 179. The induction time is the time when the PE has decreased by half of the total decrease observed during the phase transition. We additionally define the nucleation time ($t_{nuc}$) as the time of maximum rate of change of the curvature of the PE (i.e. the extremum, before the induction time, of the 3rd derivative of the fit to the PE). Because the size of the critical nucleus (and even the proper reaction coordinate) are unknown, the nucleation time serves as an estimate of the time at which a stable hydrate nucleus first forms in each nucleation trajectory. By stable hydrate nucleus, we mean the nucleus

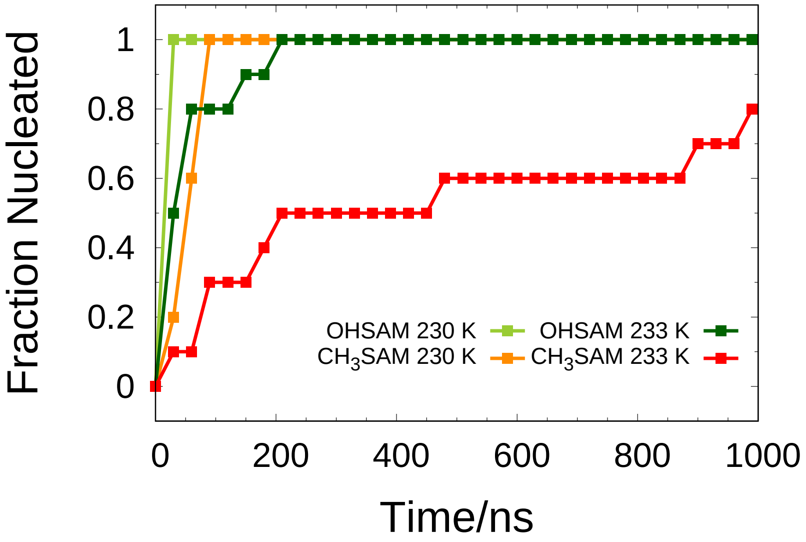

Figure 2.2: Number of trajectories nucleated after a given time. OHSAM systems at 230 K and 233 K are shown in light green and dark green, respectively. $CH_3SAM$ systems at 230 K and 233 K are shown in orange and red, respectively.

which eventually grows into hydrate, rather than one of the many nuclei that form and dissolve rapidly prior to nucleation. It is useful to know the time at which the stable hydrate nucleus forms when investigating the time evolution of system properties related to nucleation, because changes in such properties can be classified as occurring before, during, or after the formation of the stable hydrate nucleus. To ascertain the accuracy of the nucleation time, we repeated the procedure twice, replacing the PE trace with traces of two local order parameters (the largest clusters as determined by mutually coordinated guest [191] and the criteria of Báez and Clancy [192]) that quantify the size of the largest hydrate nucleus. The nucleation times calculated with the three metrics were generally within 1 ns of one another.

The induction time and nucleation time are related to the nucleation rate since the nucleation rate is the number of nucleation events volume$^{-1}$ time$^{-1}$ (or events area$^{-1}$ time$^{-1}$ in heterogeneous nucleation). All simulations reported in this work contain the same mW–XL solution volume and SAM surface area, so we report nucleation times rather than explicitly calculating nucleation rates. However, the terminology is interchangeable. On average, shorter nucleation times indicate a higher nucleation rate while longer nucleation times indicate a lower nucleation rate.

The fraction of trajectories nucleated after a given time is reported for OHSAM and $CH_3SAM$ systems in Figure 2.2. At 230 K, OHSAM systems nucleate faster than $CH_3SAM$ systems. All five

OHSAM trajectories nucleated within 50 ns, and all five $CH_3SAM$ trajectories nucleated within 100 ns. Due to the stochastic nature of nucleation, the relatively small sample size (5 trajectories for each surface), and the relatively small difference in the nucleation times ($<$ 1 order of magnitude), we wanted to ensure a statistical difference existed in the average nucleation time of OHSAM and $CH_3SAM$ systems. To this end, we performed 10 simulations of OHSAM systems and 10 simulations of $CH_3SAM$ systems at 233 K. These conditions of reduced supercooling (higher temperature) increase the barrier to nucleation, lowering the nucleation rate and accentuating any differences between OHSAM and $CH_3SAM$ systems. At 233 K, all 10 OHSAM trajectories nucleated within 200 ns, but only 8/10 $CH_3SAM$ trajectories had nucleated after 1 $\mu$s of simulation time. Together, the results at 230 K and 233 K suggest that OHSAM promotes hydrate nucleation relative to $CH_3SAM$.

### 2.3.2 Location of hydrate nucleation

There have been several proposed mechanisms of heterogeneous hydrate nucleation. Some studies have observed hydrate nucleation directly on silica surfaces, with full or half-cages hydrogen-bonding to surface -OH groups [115, 116]. Other studies have observed hydrate nucleation on an intermediate water layer which forms at silica surfaces [114, 117]. Other studies yet have found that hydrate nucleation occurs slightly away from an interface [102]. To understand how OHSAM and $CH_3SAM$ surfaces affect hydrate nucleation we first quantify the location of nucleation, i.e., the physical location in the system where the stable hydrate nucleus initially forms and begins to grow. Specifically, we are interested in the location of nucleation with respect to the SAM surfaces.

In order to accurately identify hydrate nuclei, water molecules which belong to the largest hydrate nucleus were identified every 1 ps with a local order parameter. Local order parameters are frequently used in simulation studies of crystal nucleation to identify and characterize the crystal nucleus that forms during the phase transition [193, 17, 194, 191, 104]. We use the dihedral order parameter (DHOP). DHOP is an order parameter that we created and have identified as the best approximation of the reaction coordinate for homogeneous nucleation of the XL guest [131]. DHOP calculates the dihedral angles formed from chains of neighboring water molecules. Water molecules which belong to the central bond of 11–12 planar dihedrals are considered hydrate-like. The hydrate nucleus is identified as the largest cluster of first neighbor hydrate-like water molecules.

Visualization of the trajectories suggests that the stable hydrate nuclei form at least 1 nm away from the surfaces in OHSAM and $CH_3SAM$ systems. Snapshots of nucleation in representative

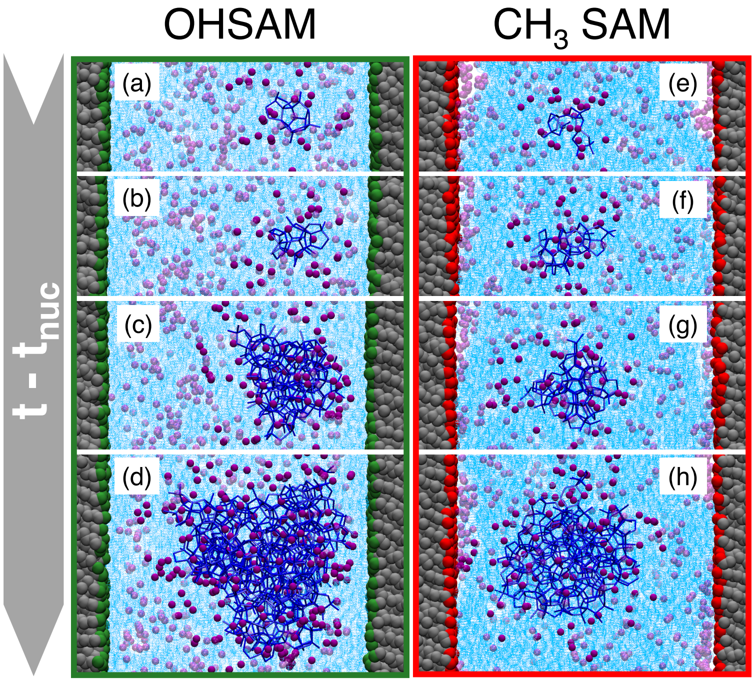

Figure 2.3: Snapshots of nucleation from representative trajectories for (a)–(d) OHSAM, and (e)–(h) $CH_3$SAM systems at 233 K. SAM surfaces are shown as gray spheres with terminal groups shown as dark green (OHSAM) or red ($CH_3$SAM) spheres. Water molecules in the largest hydrate nucleus are shown with dark blue bonds, and all other water molecules are shown as light blue points. XL guest molecules are shown as purple spheres. Snapshots (a)–(d) are from OHSAM run 5 at $t - t_{\mathrm{nuc}}/\mathrm{ns} = -1.2, -0.9, 0.0,$ and $1.2$, respectively. Snapshots (e)–(h) are from $CH_3$SAM run 6 at $t - t_{\mathrm{nuc}}/\mathrm{ns} = 0.8, 1.3, 1.5,$ and $2.1$, respectively.

trajectories from OHSAM and $CH_3$SAM systems at 233 K are shown in Figure 2.3. The snapshots are taken to show the bulk solution (light blue) between periodic images of the SAM surfaces (gray). The stable hydrate nuclei (panels (a) and (e)) initially form >1.0 nm from the terminal groups of the SAM surfaces (green and red spheres for OHSAM and $CH_3$SAM, respectively). The early nuclei are generally composed of face-sharing $5^{12}$ or half $5^{12}$ ($5^6$) type structures, similar to our observations of homogeneous nucleation of XL hydrates [131]. The hydrate nuclei initially grow slowly, undergoing structural rearrangements over the period of several hundred picoseconds (panels (a)–(b) and (e)–(f)), before growing more rapidly (panels (b)–(d) and (f)–(h)). The hydrate nuclei appear to grow in a roughly symmetric fashion until growth is slowed by crowding from the interface. The resulting hydrate is amorphous, as observed in almost all simulation studies of hydrate nucleation. It has been shown that simulations of nucleation at lower supercooling or in the microcanonical ensemble results in more crystalline hydrate [103, 109]. We note that our studies are performed at large supercooling as the melting temperature of XL hydrate is $\sim$307 K [95].

Though visualization provides insights into the location of hydrate nucleation, it is not quantitative, and furthermore, it is difficult to condense results from multiple nucleation trajectories with visualization alone. In order to quantify the location of nucleation, we calculated the local hydrate density as a function of distance from the center of the SAM surfaces ($z_{CS}$). The local hydrate density is defined as the number density of water molecules which belong to the largest hydrate nucleus. Heat maps can then be used to visualize the temporal and spatial evolution of the hydrate nucleus. In Figure 2.4 a subset of these heat maps are shown for OHSAM and $CH_3$SAM at 233 K. The results are reported for 5 ns centered on the nucleation time of each trajectory. We show the extent of the SAM surfaces with green and red crosshatched rectangles for OHSAM and $CH_3$SAM, respectively. In all cases the hydrate density is initially low and fluctuates from $\sim$0.5 nm to $\sim$3.0 nm from the SAM surfaces. Examples of this behavior can be seen in most of the trajectories, but specifically from $t - t_{\text{nuc}} = -2.5$ ns to $t - t_{\text{nuc}} = -1.5$ ns in panel (c) and from $t - t_{\text{nuc}} = -2.5$ ns to $t - t_{\text{nuc}} = 0.5$ ns in panel (i). During this period, the largest nucleus frequently changes location in the system, indicating that a stable nucleus has not formed. Once the stable hydrate nucleus forms, it remains in one location and begins to grow. This causes the density of the hydrate to increase at a given $z$-location. Thus, the heat maps provide a quantitative picture of where in the system the stable hydrate nucleus forms with respect to the SAM surfaces. The heat maps highlight the uncertainty in our nucleation time estimates. In some trajectories the stable hydrate nucleus forms before $t_{\mathrm{nuc}}$, while in other trajectories it forms after $t_{\mathrm{nuc}}$. In almost all cases the stable nucleus forms within 2 ns of $t_{\mathrm{nuc}}$.

For all nucleation trajectories in OHSAM and $CH_3$SAM systems it is clear that hydrate nuclei never form directly on the SAM surfaces. Even prior to the formation of a stable hydrate nucleus, when the location of the largest nucleus is rapidly fluctuating throughout the system, water molecules within 0.5 nm of the SAM surface rarely belong to hydrate nuclei. At the time that the stable nucleus forms, the closest edge of the nucleus is most often further than 1.0 nm from the SAM surface. Furthermore, the regions of highest hydrate density first appear several nanometers from the SAM surface. These results support our observations in Figure 2.3. Similar behavior is also observed for all trajectories at 230 K. We find that the stable hydrate nuclei do not form at OHSAM or $CH_3$SAM surfaces. Thus, hydrate nucleation is not directly templated by either hydrophilic or hydrophobic SAM surfaces.

### 2.3.3 Spatial distribution of water and guest molecules

Next, we investigated how the presence of SAM surfaces affected the spatial distribution of water and guest molecules. It is well-established that guest concentration has a large effect on hydrate nucleation rates [68, 67, 195, 98]. The density of SAM, mW, and XL were calculated as a function of distance from the center of the SAM surfaces during the pre-nucleation period. The pre-nucleation period is defined as starting 5 ns into the production nucleation simulations and ending 5 ns before the nucleation time. This definition ensures that averages over this period represent properties of the metastable solution, and are unaffected by hydrate nucleation. We focus our analysis on the pre-nucleation period to elucidate characteristics of the metastable solution responsible for hydrate nucleation.

Figure 2.5 reports the density of SAM, mW, and XL in OHSAM and $CH_3$SAM systems at 233 K. Water forms two distinct peaks near OHSAM indicating water layering near the hydrophilic surface. Very little XL is found within 0.5 nm of OHSAM. The XL density reaches a maximum at $\sim$1.0 nm from the OHSAM surface before reaching a plateau at $\sim$2.0 nm from OHSAM. We refer to the XL density and concentration in the plateau region as the bulk density and bulk concentration of XL, respectively. The maximum XL density is $\sim$1.7 times the bulk XL density in the OHSAM systems.

The distribution of water and XL is notably different in $CH_3$SAM systems. The first peak

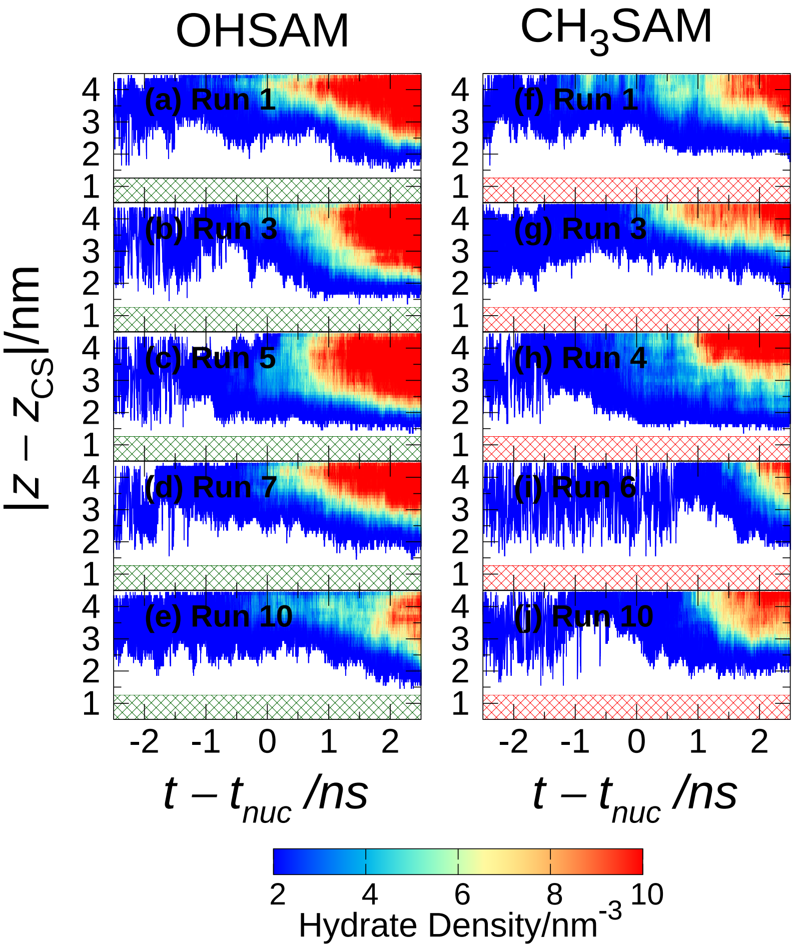

Figure 2.4: Local hydrate density as a function of time and distance from the SAM for five representative nucleation trajectories in OHSAM (panels (a)–(e)) and $CH_3$SAM (panels (f)–(j)) systems at 233 K. $|z - z_{\mathrm{CS}}|$ is the distance from the center of the SAM. The location of the SAM surfaces are shown as green (OHSAM) and red ($CH_3$SAM) crosshatched rectangles. Results are reported for 5 ns centered on the nucleation time of each trajectory.

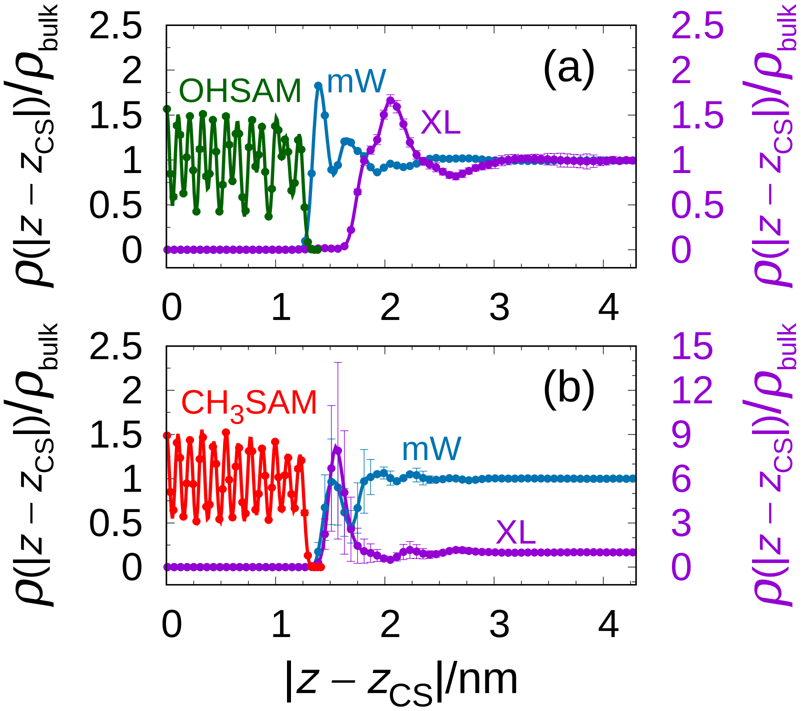

Figure 2.5: Normalized density of SAM, mW (blue), and XL (purple) as a function of distance from the center of the SAM surfaces for (a) OHSAM and (b) CH$_3$SAM systems at 233 K. OHSAM is shown in green and CH$_3$SAM is shown in red. SAM and mW densities are reported on the primary y-axis, and XL density is reported on the secondary y-axis. Densities were calculated prior to nucleation and averaged across all runs. Error bars represent one standard deviation from the mean. Bulk densities are averaged from $0.0 < |z - z_{\mathrm{cs}}|/\mathrm{nm} < 1.0$ for SAM and $3.0 < |z - z_{\mathrm{cs}}|/\mathrm{nm} < 4.0$ for mW and XL. In OHSAM systems $\rho_{\mathrm{bulk,mW}} = 29.2$ molecules/nm$^3$ and $\rho_{\mathrm{bulk,XL}} = 1.96$ molecules/nm$^3$. In CH$_3$SAM systems $\rho_{\mathrm{bulk,mW}} = 30.4$ molecules/nm$^3$ and $\rho_{\mathrm{bulk,XL}} = 1.27$ molecules/nm$^3$.

in the water density is further from the $CH_3$SAM surface compared with OHSAM, a feature characteristic of hydrophobic interfaces [173, 171, 172, 196, 197, 198, 199]. The water density in the first peak is considerably lower for $CH_3$SAM compared with OHSAM. The second peak in the water density is almost nonexistent. The lower water density near the $CH_3$SAM surface may be attributed to the presence of XL, which aggregates at the $CH_3$SAM surface. The maximum in the XL density is located only 0.3 nm from $CH_3$SAM and is ∼8 times the bulk XL density in the $CH_3$SAM systems. The error bars on mW and XL density are large near the $CH_3$SAM surface because unequal amounts of XL aggregated at the two $CH_3$SAM interfaces (see Figure 2.3). It appears that once XL begins to aggregate at one interface, more XL follows, causing a pseudo-phase separation. We note that the average density profiles of water and guest were approximately constant during the pre-nucleation period.

### 2.3.4 Time evolution of water structure

Though the stable hydrate nucleus clearly forms >1.0 nm from the SAM surfaces, the surfaces could be inducing interfacial water structure which leads to nucleation. During the nucleation of $CO_2$ hydrates in the presence of silica surfaces, researchers observed the formation of an ice-like layer on the silica surfaces which preceded nucleation of clathrate hydrate [114]. Other studies have additionally reported hydrate nucleation near surfaces with some intermediate water layer bridging the surface and hydrate [117].

To investigate if similar phenomena occurred in our systems, the time evolution of water structure was characterized in $z$-slabs at different distances from the surfaces. The bounds of the $z$-slabs were chosen to coincide with the minima in the water density profiles shown in Figure 2.5. The water structure was characterized with an angular order parameter, $F_\theta$ [192], which is approximately zero for perfectly tetrahedral structure, and a dihedral order parameter, $F_{4\phi}$ [200], which is often used to distinguish liquid, ice, and hydrate structure in water. An average $F_\theta$ is used:

$$F_\theta = \left\langle \sum_{i=1}^{n_{\text{ang}}} (|\cos x_i| \cos x_i + 0.11)^2 \right\rangle_z \tag{2.4}$$

where $\langle \cdots \rangle_z$ indicates an average over all water molecules in the $z$-slab of interest, $n_{\text{ang}}$ is the number of angles formed between a water molecule and its first neighbors and $x_i$ is the value of angle $i$. First neighbors are defined as water molecules within 0.325 nm of each other. SAM terminal groups within

0.325 nm of a water molecule are also included as a first neighbor. For reference, $F_\theta$ of liquid water is $\sim 0.4$ and $F_\theta$ of solid water (ice or hydrate) is $\sim 0.05$. A modified version [105] of the $F_{4\phi}$ [200] order parameter is used:

$$F_{4\phi} = \frac{1}{n_d} \sum_{i} \sum_{j \neq i} \sum_{k \neq i,j} \sum_{l \neq i,j,k} \cos(3\theta_{i,j,k,l}). \tag{2.5}$$

$\theta_{i,j,k,l}$ is the dihedral angle between four neighboring water molecules (i.e. $i$ and $j$, $j$ and $k$, and $k$ and $l$ must be first neighbors). $n_d$ is the total number of dihedrals in the $z$-slab. Once again, SAM terminal groups are included as first neighbors of a water molecule if they meet the distance criteria. $F_{4\phi}$ of liquid water is $\sim$0.0, ice is $< -0.4$, hydrate is $\sim$0.9.

The time evolution of $F_\theta$ and $F_{4\phi}$ are shown in Figure 2.6 for OHSAM and $\mathrm{CH_3}$SAM systems at 233 K. The results are reported as an average of all trajectories that nucleated. The water molecules located in $z$-slabs closest to the SAM surfaces (gray and orange) show no change in structure, as characterized by $F_\theta$ (panels (a) and (c)) or $F_{4\phi}$ (panels (b) and (d)), before, during, or after nucleation. Furthermore, these water molecules are well within the region defined as liquid-like for both $F_\theta$ and $F_{4\phi}$. $F_\theta$ of water closest to the $\mathrm{CH_3}$SAM surface (panel (c), gray), suggests increased tetrahedriality. However, we believe this signal arises from the lower number of first neighbors for water molecules near a hydrophobic surface. The lack of any difference between $F_{4\phi}$ of water in the $z$-slab closest to $\mathrm{CH_3}$SAM (panel (d), gray) and OHSAM (panel (b), gray) supports this hypothesis.

The first changes in water structure captured by $F_\theta$ and $F_{4\phi}$ occur in the $z$-slabs furthest from the SAM surfaces (blue and magenta lines). These changes generally occur around the nucleation time and are attributed to the formation and growth of the stable hydrate nucleus. We find no evidence of intermediate structure (ice-like or otherwise) forming on the SAM surfaces prior to hydrate nucleation. In-plane mW–mW, XL–XL, and mW–XL radial distribution functions (not reported) averaged over different time windows in the pre-nucleation period also showed no changes in the water or guest structure vicinal to SAM surfaces prior to nucleation.

### 2.3.5 Nucleation in homogeneous systems

Since there is no evidence that hydrate nucleation occurred at or near the SAM surfaces, and no evidence of any intermediate structure formation on the SAM surfaces, it is possible that the only manner in which the SAM surfaces influenced hydrate nucleation is by affecting the guest

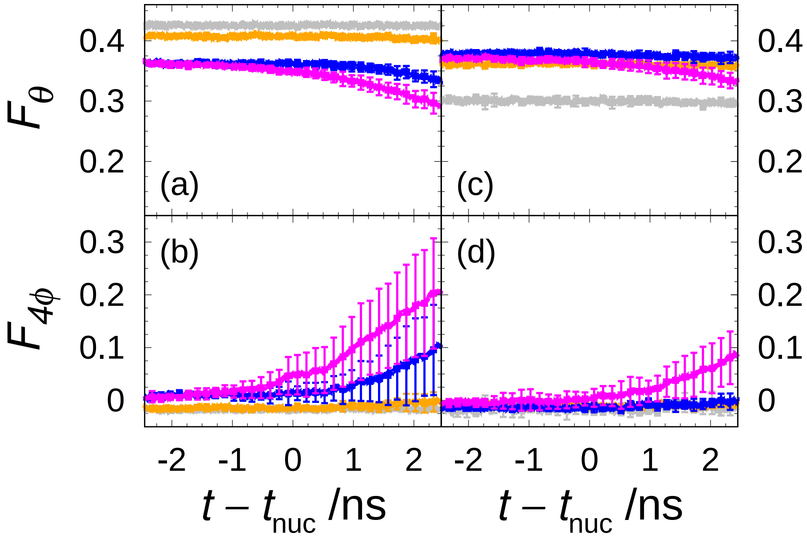

Figure 2.6: Evolution of water structure as characterized by angular ($F_\theta$) and dihedral ($F_{4\phi}$) parameters in $z$-slabs of different distances from the (a)–(b) OHSAM and (c)–(d) CH$_3$SAM surfaces near the nucleation time. The bounds of the $z$-slabs were selected to match the minima of the water density profiles shown in Fig. 2.5. Gray, orange, blue, and magenta lines represent slabs corresponding to $|z - z_{\text{cs}}|/\text{nm} = 1.2$–$1.53$, $1.53$–$1.93$, $1.93$–$2.50$, and $2.50$–$4.00$, respectively, for OHSAM systems, and $|z - z_{\text{cs}}|/\text{nm} = 1.31$–$1.69$, $1.69$–$2.11$, $2.11$–$2.50$, and $2.50$–$4.00$, respectively, for CH$_3$SAM systems. Results are averaged with a 10 ps rolling average for each run and then are averaged across all runs that nucleated at 233 K. Error bars represent one standard deviation from the mean.

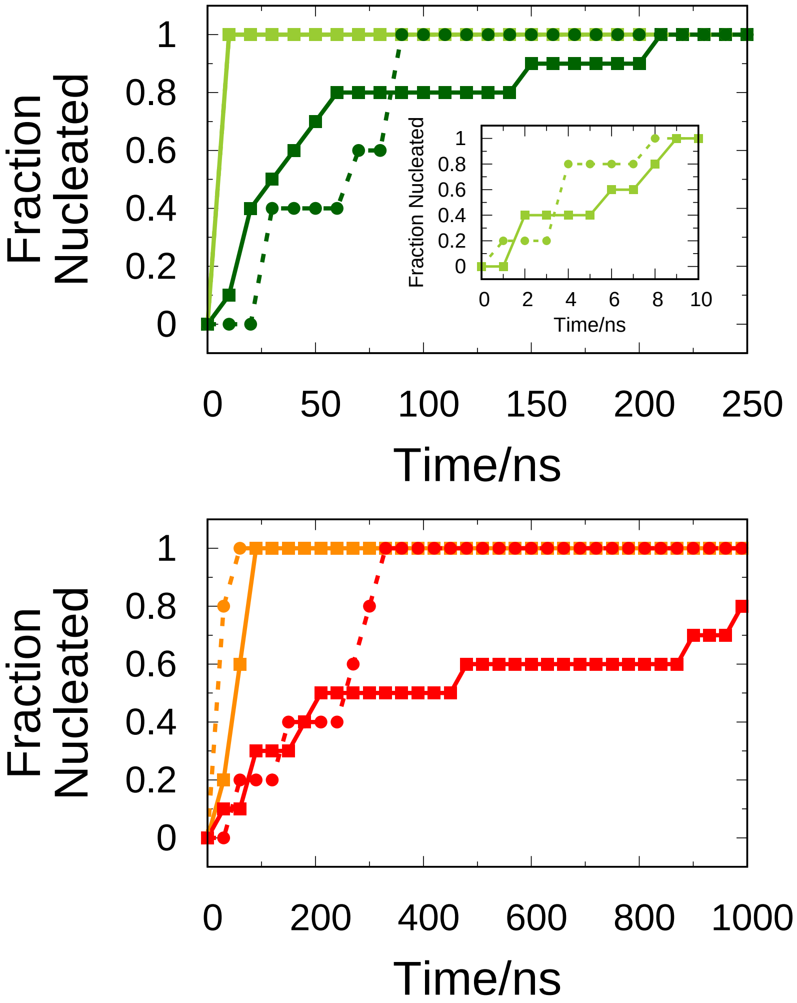

Figure 2.7: Fraction of trajectories nucleated after a given time for bulk homogeneous solution of mW and XL compared with the (a) OHSAM and (b) $CH_3$SAM systems. Systems with SAM are shown as points with a solid line and homogeneous systems are shown as points with a dashed line. Light and dark green lines are OHSAM at 230 K and 233 K, respectively. Orange and red lines are $CH_3$SAM at 230 K and 233 K, respectively. The concentrations of the bulk systems were selected to match the bulk concentration of XL in the OHSAM or $CH_3$SAM systems. $x_{\mathrm{bulk}} = 6.5$ mol% XL for OHSAM and 4.0 mol% XL for $CH_3$SAM.

distribution in the systems. Though the OHSAM and $CH_3$SAM systems have the same number of mW and XL molecules, the expulsion of XL near the OHSAM surface and the aggregation of XL near the $CH_3$SAM surface (see Figure 2.5) results in different bulk XL concentrations for OHSAM (6.5 mol%) and $CH_3$SAM (4.0 mol%) systems. If nucleation in the SAM systems is homogeneous (i.e. not directly affected by the presence of the SAM surfaces), the nucleation rates in the SAM systems should be similar to homogeneous systems of mW and XL. In order to compare nucleation rates observed in the SAM systems with homogeneous nucleation rates at equivalent concentrations, simulations of mW–XL solutions in the absence of any SAM surface at 4.0 mol% and 6.5 mol% XL were performed.

Five nucleation simulations were performed for each of the homogeneous systems (4.0 mol% XL and 6.5 mol% XL) at 230 K and 233 K. The nucleation times in the homogeneous systems are compared with the nucleation times in the SAM systems in Figure 2.7. The results for OHSAM and the equivalent homogeneous system (6.5 mol% XL) show that the nucleation times are similar (see panel (a) and inset) at both 230 K and 233 K. This suggests that the presence of the OHSAM surface does not influence nucleation except for how it affects the distribution of XL guest in the system. $CH_3$SAM and the equivalent homogeneous system (4.0 mol% XL) shows similar nucleation times at 230 K. However, at 233 K the nucleation times for the $CH_3$SAM systems are notably longer than homogeneous systems with an equivalent bulk XL concentration. This suggests that the presence of the $CH_3$SAM surface hinders hydrate nucleation.

### 2.3.6 Formation of XL contact pairs near $CH_3$SAM surface

How does the presence of the $CH_3$SAM surface hinder hydrate nucleation? It is surprising that $CH_3$SAM hinders hydrate nucleation given that there is a $\sim$8 times increase in the local XL density near the $CH_3$SAM surface. Even if the XL concentration in the $z$-slab next to the $CH_3$SAM surface is too high for hydrate formation, the high concentration of XL in this $z$-slab could act as a reservoir of XL. Nucleation might then be promoted in a $z$-slab just outside of the high-concentration region. However, we observe the opposite effect. $CH_3$SAM appears to hinder nucleation rather than promote it, even when compared to homogeneous systems of equivalent concentration. Furthermore, Figure 2.4 shows no indication of nucleation occurring near the region of high XL concentration. Therefore, the $CH_3$SAM surface must be affecting the water–water, guest–guest or guest–water interactions in a manner which hinders hydrate nucleation, despite the increased local guest concen-

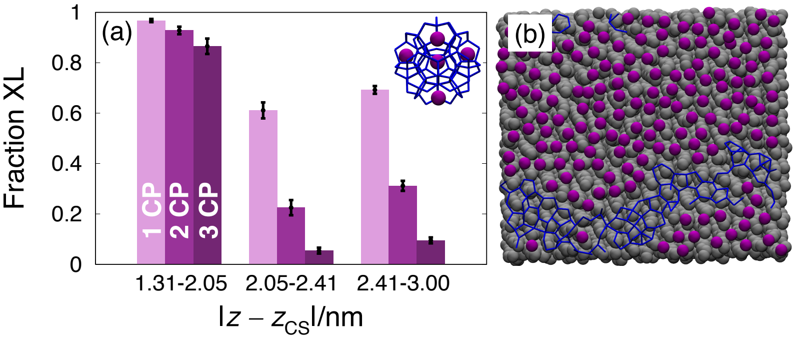

Figure 2.8: (a) Fraction of XL molecules participating in at least 1 (light purple), 2 (purple), or 3 (dark purple) contact pairs with other XL molecules in $z$-slabs at different distances from the $CH_3$SAM surface. Results are averaged across the pre-nucleation period of all trials. Error bars represent one standard deviation on the mean. A snapshot of the sII crystal is shown as an inset. (b) Snapshot of XL molecules near the $CH_3$SAM surface illustrating XL aggregation and the formation of XL contact pairs. SAM is shown in gray, XL molecules within 0.6 nm of the SAM surface are shown as purple spheres, and mW molecules within 0.6 nm of the SAM surface are shown with blue bonds.

tration.

Visualization of XL molecules near the $CH_3$SAM surface revealed that the XL molecules commonly adopted contact configurations with one another. Contact configurations are configurations where guest molecules are “in contact” with one another with no water molecules between them. A snapshot of XL near the $CH_3$SAM (Figure 2.8(b)) surface shows many such configurations. Figure 2.8(a) reports the fraction of XL molecules participating in at least 1, 2, or 3 contact pairs with other XL molecules in different $z$-slabs in the $CH_3$SAM system at 233 K. XL molecules belong to an increased number of contact pairs near the $CH_3$SAM surface. In particular, the fraction of XL molecules in 3 or more contact pairs is significantly enhanced near $CH_3$SAM. The formation of contact pairs may prevent hydrate nucleation since all guest molecules in the hydrate structure exist in solvent separated pairs for singly–occupied cages. For reference, a snapshot of guest molecules in the sII crystal is shown as an inset in Figure 2.8(a). Previous work has shown that two methane molecules near a hydrophobic surface display a shallower solvent separated minimum and lower desolvation barrier in the methane–methane potential of mean force [201]. Despite the fact that XL is a water-soluble guest molecule, our results suggest that similar behavior is observed for XL near $CH_3$SAM.

Collectively, our results indicate that the primary effect of surfaces on hydrate nucleation arises from changes in the guest concentration in the bulk solution. In our systems, we observe that the XL guest molecules aggregate at the $CH_3$SAM surface resulting in reduced XL concentration in bulk, while OHSAM expels guests close to the surface, resulting in a slight increase in the bulk XL concentration. Interestingly, the nucleation rate in OHSAM systems is similar to the homogeneous nucleation rates at the enhanced bulk concentration. Therefore, one could argue that OHSAM surfaces are "enhancing" hydrate nucleation, though not through the traditional mechanisms (e.g. templating, etc). Various reasons for hindrance or enhancement of hydrate nucleation on silica surfaces have centered around the structuring of water near the surfaces [114, 115, 116, 117]. Often times, it has been argued that either the formation or not of ice-like layers affects hydrate nucleation. In our studies, we find that neither $CH_3$SAM nor OHSAM promote such ice-like features. We note that recent studies have shown that surface flexibility often hinders ice nucleation [202, 181]. This indicates that non-crystalline and flexible surfaces such as SAM surfaces will likely show different mechanisms for hydrate (and more generally crystal) nucleation.

## 2.4 Conclusions

We studied the effect of model hydrophilic and hydrophobic surfaces on clathrate hydrate nucleation with extensive MD simulations. Since hydrate nucleation is a stochastic process, 5–10 independent nucleation trajectories were generated for every system reported in this study. SAM surfaces were selected as model surfaces because they afford a range of potential surface chemistries with no rigid crystalline structure. We studied the nucleation of XL hydrates, where XL is a water-soluble, structure II hydrate forming guest molecule. We observe that hydrate nucleation always occurred in the bulk, $>1.0$ nm from the SAM surfaces, suggesting that neither surface templates the growth of a hydrate crystal directly at the surface. Furthermore, the formation of intermediate water structure (ice-like, or otherwise) is not observed on or near the SAM surfaces prior to nucleation. Instead, we find that the presence of SAM surfaces affected the guest concentration in bulk. $CH_3$SAM promotes aggregation of XL molecules near the surface effectively reducing the bulk XL concentration. In OHSAM systems the guest concentration in bulk is increased due to guest exclusion near the surface. Therefore, the surface effects on hydrate nucleation primarily come from their effects on the distribution of guest molecules in the system rather than structuring of water near these surfaces. This highlights the interplay of various mechanisms that can affect heterogeneous nucleation in multicomponent systems such as clathrate hydrates. Extrapolating our results to realistic systems, we speculate that in systems with constant chemical potential (i.e. where an infinite bath of guest molecules is available from the bulk phase), guest will aggregate at hydrophobic surfaces resulting in a “vapor-like” layer. Nucleation may then occur at the resulting vapor–liquid interface as has been reported in the literature [97, 104].

# Chapter 3

# Nucleation mechanism of clathrate hydrates of water-soluble guest molecules[1]

## 3.1 Introduction

Current experiments lack the requisite spatial and temporal resolution to study the molecular-level details of hydrate nucleation [165, 166]. Increasing computational power and advanced sampling algorithms [123, 1, 127] have enabled molecular simulations of hydrate nucleation over the previous decade [93, 94, 27, 95, 96, 100, 99, 101, 62, 115, 102, 104, 105, 106, 117, 108, 109, 113]. Most studies have focused on sparingly soluble guest molecules (particularly $CH_4$ [93, 94, 27, 96, 100, 101, 62, 102, 104, 105, 106, 109, 107, 203] and $CO_2$ [115, 117, 113]) that form structure I (sI) hydrates. The emerging consensus indicates that nucleation of hydrates from sparingly soluble guest molecules begins with thermally-driven fluctuations that result in the formation and dissolution of pre-critical amorphous hydrate nuclei [101, 108], often within 'blobs'; aggregates of solvent-separated guest molecules with lifetimes longer than pre-critical clusters of clathrate cages which form and dissolve within them [27, 95]. The stability of these blobs may in fact arise from the presence of incomplete cage structures which are composed of edge-sharing planar water faces [92, 204]. Regardless of the

[1] Material for this chapter adapted from Ref. [131]

interplay of water and guest structuring which results in long-lived blobs, complete cages form from within this region. The resulting hydrate nuclei are composed of complete (and incomplete) polyhedral cages but have no long-range crystalline order. When a hydrate nuclei reaches post-critical size, it grows rapidly, exhibiting a combination of sI, structure II (sII), and other structural motifs. It is not difficult to imagine that the early stages of nucleation of soluble and sparingly soluble guest molecules might be different due to differences in their water-mediated interactions. Indeed, blobs formed by soluble guest molecules are not as long-lasting as blobs formed by sparingly soluble guest molecules [95].

A handful of recent studies have investigated the nucleation of clathrate hydrates of water-soluble [112] and mixed guests [111, 205, 110]. The results suggest that water-soluble and mixed guests also form amorphous hydrate structures, similar to the nucleation of hydrates of sparingly soluble guests. However, the description of the nucleation mechanism of hydrates of soluble guests remains incomplete. Though less studied in molecular simulations, water-soluble guest molecules have important technological applications. For example, there have been recent advances utilizing THF as a promoter molecule for long-term natural gas storage in sII hydrates [81, 87, 82]. Other researchers have explored using THF as a promoter molecule for hydrogen storage in hydrate form [84, 81, 87, 82]. Engineering these technological applications requires fine-tuning hydrate forming conditions. This necessitates a complete understanding of the nucleation of mixed hydrate systems. Rigorously investigating and quantifying the mechanism of nucleation of hydrates of water-soluble guest molecules represents one of the important steps towards this goal.

In this work we study the mechanism of nucleation of clathrate hydrates of a water-soluble guest molecule using a combined forward flux sampling [129, 1] (FFS)–committor probability analysis [144]. We identify order parameters (OPs) that provide the best approximation of the reaction coordinate and characterize the transition state (TS). Our calculations represent the most comprehensive effort to date to compare various OPs which have been proposed to study hydrate nucleation of both sparingly soluble and soluble guest molecules. We use insights gained from the reaction coordinate and TS analysis to motivate further simulations which probe molecular level details of the nucleation mechanism.

## 3.2 Methods

We studied the nucleation of clathrate hydrates from a homogeneous solution of monatomic water [174] (mW) and the XL guest molecule [95] in the $NpT$ ensemble at 5.6 mol% XL, $p = 500$ atm, and $T = 230$ K. The XL guest is water-soluble, approximately the size of THF, and occupies the $5^{12}6^4$ cages of sII hydrate [95]. 5.6 mol% is the concentration of XL required to occupy all $5^{12}6^4$ cages of sII hydrate. We note that since XL is a water-soluble guest molecule, our simulations are not at artificially high concentration; experiments of mixed-THF hydrate formation are sometimes performed at similar THF concentrations [87, 206].

### 3.2.1 Forward flux sampling

Direct forward flux sampling was performed to generate a large number of nucleation pathways. FFS is a technique used to sample rare event transitions by propagating a system from an initial state, or basin, to a final state, through a series of successive interfaces ($\lambda_0, \lambda_1 ... \lambda_n$) defined by increasing values of some OP, $\lambda$ [129, 1]. Once FFS is complete, transition pathways are constructed by connecting trajectories backwards from $\lambda_n$ to $\lambda_0$. The flux of trajectories from the initial basin of attraction, $\lambda_A$, to $\lambda_0$ is calculated from straightforward MD simulations in $\lambda_A$. These basin simulations are also used to collect system configurations at $\lambda_0$. The rate constant for the process can then be calculated as $k_{FFS} = \Phi_0 \prod_{i=0}^{n-1} P(\lambda_{i+1}|\lambda_i)$, where $\Phi_0$ is the flux of trajectories from $\lambda_A$ to $\lambda_0$ and $P(\lambda_{i+1}|\lambda_i)$ is the probability that a trajectory initiated from $\lambda_i$ reaches $\lambda_{i+1}$ before returning to $\lambda_A$. Further details of the direct FFS algorithm can be found in Ref. 1. The OP used for FFS was $\text{BC}_{\text{planar}}$. $\text{BC}_{\text{planar}}$ is a local OP which identifies the largest cluster of hydrate-like water molecules based upon the rules of Báez and Clancy [192] (see Table 3.1 for a detailed description). We selected $\text{BC}_{\text{planar}}$ as the OP for FFS because previous work has shown that hydrate nuclei are amorphous and the early stages of nucleation may involve partial cage structures. $\text{BC}_{\text{planar}}$ identifies hydrate-like structure regardless of crystal structure (sI/sII) and does not require that a water molecule be a part of a complete cage.

One hundred independent 5 ns basin simulations were performed to calculate the flux of trajectories from the $\lambda_A$ to $\lambda_0$, and to collect configurations at $\lambda_0$. Initial configurations for the basin simulations were generated in the following manner: coordinates for mW and XL molecules were randomly generated at the appropriate density. This configuration was equilibrated at $T = 300$

K and $p = 1$ atm for 1 ns. The pressure was increased to 500 atm and the system was simulated in production for 25 ns. Configurations were saved every 1 ns from 16–25 ns. Ten independent simulations were initiated from each of the ten configurations by initializing each simulation with randomly assigned velocities consistent with the Maxwell-Boltzmann distribution at $T = 230$ K. Each simulation was allowed to equilibrate for 1 ns at 500 atm and 230 K, followed by 5 ns of data collection. The boundary of the liquid basin, $\lambda_A$, was selected as the mean cluster size in the liquid basin. $\lambda_0$ was selected as two standard deviations beyond $\lambda_A$. Details of the flux calculation are reported in Appendix A. First crossings of $\lambda_0$ separated by at least 500 ps were selected as initial configurations for FFS to prevent correlation between successive configurations at $\lambda_0$. This procedure resulted in a total of 778 configurations at $\lambda_0$.

FFS was performed with 10,000–40,000 total simulations per interface. The value of each interface, $\lambda_i$ was chosen on-the-fly based on the progress of trajectories initiated from the previous interface, $\lambda_{i-1}$. The FFS simulations were performed using SAFFIRE. SAFFIRE is a software package under development in our group which uses the Hadoop platform [207, 208] and concepts from SciFlow [209] to manage the submission and analysis of large numbers of individual MD simulations. FFS details, including interface values, the number of configurations harvested at each interface, the total number of trajectories initiated from each interface, the number of trajectories that successfully crossed the next interface, and the probability of advancing to the next interface, are reported in Appendix A. In total, FFS generated 1101 transition paths. Interestingly, we observed that 1021 transition paths originated from a single configuration at the first interface. We refer to this configuration hereinafter as C753. Despite the success of pathways originating from C753, no single characteristic difference between C753 and other configurations at the first interface was identified. Further discussion is provided in Appendix A.

### 3.2.2 Committor analysis and model fitting

The committor probability, $p_{\mathrm{B}}(\mathbf{x})$, is the probability that a nucleus will grow to form hydrate. Though it contains no mechanistic or structural information, the committor probability is sometimes considered the perfect reaction coordinate since it describes the progress of the transition [210, 211, 135]. Prospective reaction coordinate models can be compared by fitting each model to $p_{\mathrm{B}}(\mathbf{x})$ for a collection of configurations representative of the transition path ensemble (TPE), and evaluating goodness-of-fit [144]. Better reaction coordinate models will have a better fit to the $p_{\mathrm{B}}(\mathbf{x})$ data. In other words, a good reaction coordinate model should be able to accurately predict the committor probability of any configuration which belongs to the TPE.

The committor probability of each configuration from FFS, $p_{\mathrm{B,FFS}}(\mathbf{x})$, was initially estimated from the connectivity of the trajectories [144]. The accuracy of $p_{\mathrm{B,FFS}}(\mathbf{x})$ was tested by randomly selecting 20 configurations and calculating their committor probability with 100 straightforward MD simulations ($p_{\mathrm{B,MD}}(\mathbf{x})$). The difference between $p_{\mathrm{B,FFS}}(\mathbf{x})$ and $p_{\mathrm{B,MD}}(\mathbf{x})$ was as large as 0.55 for some configurations. The differences between $p_{\mathrm{B,FFS}}(\mathbf{x})$ and $p_{\mathrm{B,MD}}(\mathbf{x})$ arise from the uncertainty in $p_{\mathrm{B,FFS}}(\mathbf{x})$. Initiating even 30 trajectories from a single configuration can result in one standard deviation on the estimate as large as 0.10. Furthermore, $p_{\mathrm{B,FFS}}(\mathbf{x})$ values at earlier interfaces are calculated by averaging over $p_{\mathrm{B,FFS}}(\mathbf{x})$ from later interfaces, further propagating and compounding any error in the estimates. Refer to Ref. 144 for a complete description of the method.

Since $p_{\mathrm{B,FFS}}(\mathbf{x})$ values appeared inaccurate, accurate committor probabilities were calculated for 153 configurations sampled during FFS with straightforward MD simulations. The selection of these configurations was motivated by two goals: (1) to evenly distribute the $p_{\mathrm{B}}(\mathbf{x})$ values between $p_{\mathrm{B}}(\mathbf{x}) = 0.0$ and $p_{\mathrm{B}}(\mathbf{x}) = 1.0$ to prevent overfitting to one region of the transition, and (2) to compare the transition mechanism for paths originating from C753 with the transition mechanism for other paths. Within the framework of these two goals, the configurations were randomly selected. For each selected configuration 100–200 MD simulations were initiated with randomly generated velocities consistent with the Maxwell-Boltzmann distribution. The simulations were continued until they committed to the liquid or solid basin to yield $p_{\mathrm{B}}(\mathbf{x})$ with $2\sigma < \pm 0.07$. Our final collection of 140 configurations with $0.0 < p_{\mathrm{B,MD}}(\mathbf{x}) < 1.0$ consisted of 76 configurations belonging to transition paths originating at C753 and 64 configurations belonging to other transition paths.

Two sets of configurations were created from the 140 configurations with $0.0 < p_{\mathrm{B,MD}}(\mathbf{x}) < 1.0$. The first set, TP-TPE, was comprised of the 76 configurations belonging to transition paths initiated at C753 and 6 configurations belonging to transition paths beginning from other configurations at the first interface. The TP-TPE set of configurations is consistent with the TPE as sampled by FFS [129, 1]. The second set, TP-NC753, was comprised of 64 configurations that belong to transition paths not originating from C753. The fact that 93% of the transition paths originate from C753 suggests that this configuration is extremely reactive compared with the other configurations sampled at the first interface. Though many configurations at the first interface spawned trajectories that progressed through several interfaces of FFS, the majority of these trajectories eventually returned to the liquid basin. As such, successful transition pathways that do not originate from C753 represent less traveled (and presumably higher energy) transition pathways. Fitting reaction coordinate models to the two different sets of configurations enables comparison of the transition pathways.

Prospective reaction coordinate models were created from single OPs and linear combinations thereof. Each prospective model was fit to the two different sets of configurations, TP-TPE and TP-NC753. Reaction coordinate models with up to five OPs were identified with a forward and backward stepwise procedure. At each step, the Bayesian Information Criterion (BIC) was used to determine whether a parameter should be added or removed from the model. BIC rewards improved fit while penalizing additional model complexity [212]. Lower values of BIC indicate a better model. Due to concerns about over fitting and difficulty in physically interpreting the results, our discussion focuses on single parameter models. The most important predictors in the models identified from the stepwise BIC procedure were used to create the two parameter models reported in Table 3.3. Prospective single parameter models were ranked with the model $R^2$ and models with different numbers of parameters were compared with BIC. A cross interaction term was initially included in all two parameter models, but an ANOVA analysis determined that the cross term was nearly always insignificant. We note that the cross interaction term was insignificant for the two-parameter models reported in Table 3.3.

### 3.2.3 Order parameters

The primary classes of OPs tested were as follows: OPs based on the work of Báez and Clancy [192] (BC), the mutually coordinated guest (MCG) OPs from Barnes *et al.* [191], the half cage OP from Bi and Li [104] (HCOP), the largest solvent separated guest (LSSG) OP from Jacobson *et al.* [194], complete cage-based OPs, OPs based upon the face-saturated incomplete cage analysis (FSICA) from Guo *et al.* [204], and a novel OP based upon identifying planar dihedrals (DHOP). Previous studies have used planar water structures as a signature of hydrates; examples include the planar percentage of primitive rings [213] and CHILL+ [214]. Abbreviations and definitions for the 33 OPs evaluated are provided in Table 3.1.

Since DHOP is a novel OP, the algorithm is described in detail here. DHOP is defined as the size of the largest cluster of hydrate-like water molecules, where hydrate-like water molecules are identified with the following procedure:

Table 3.1: List of OPs evaluated in this study

| Abbreviation | Description |
|---|---|
| BC | Original Báez and Clancy identification [192]. A water molecule must be part of 4–6 5-membered rings (5MRs) and have tetrahedral order. |
| $\mathrm{BC}_{\mathrm{planar}}$ | BC with angle criteria ($> 60°$) and dihedral criteria ($< 30°$) to identify planar 5MRs. |
| $\mathrm{BC}_{56}$ | BC modified to require a water to be part of 5–6 5MRs rather than the original 4–6 5MRs. |
| $\mathrm{MCG}_x$ | Original mutually coordinated guest (MCG) OP [191] where a guest molecule must be part of $>= x$ mutually coordinated pairs to be considered an MCG monomer. A mutually coordinated pair of guests is defined as two guest molecules with $>= 5$ water molecules in a region defined by two overlapping cones emanating from the guest molecules and oriented along the guest–guest vector. $x$ =1,2, or 3. |
| $\mathrm{MCG\text{-}6R}_x$ | MCG OP modified to require 6 water molecules in the region between the two guests for them to be considered an MCG pair. A guest must be part of $>= x$ mutually coordinated pairs to be considered an $\mathrm{MCG\text{-}6R}_x$ monomer. $x$ =1,2, or 3. |
| $\mathrm{BC\text{-}MCG}_x$ | Water molecule must satisfy BC criteria and be a first neighbor of a guest molecule which is an $\mathrm{MCG}_x$ monomer. $x$ =1,2, or 3. |
| LSSGOP | Solvent-separated guest OP of Jacobson *et al.* [194] Identifies guest molecules at a certain distance as solvent-separated. |
| HCOP | Half-cage OP of Bi *et al.* [104]. Identifies the largest cluster of water molecules in face-sharing half ($5^6$, $5^6 6^1$, $5^6 6^2$) or full cages ($5^{12}$, $5^{12}6^2$,$5^{12}6^3$,$5^{12}6^4$) |
| $F_{4\phi}$-$y$ | $F_{4\phi}$ [200] averaged over the largest cluster identified by OP $y$, where $y$ = BC, $\mathrm{BC}_{\mathrm{planar}}$, or $\mathrm{BC}_{56}$ |
| $\mathrm{DHOP}_{x^\circ}$ | Novel OP that requires that a water molecule belong to 11-12 planar dihedrals with its first neighbors to be considered hydrate-like, where a planar dihedral has an angle $<= x$. Further details provided in the text. |
| RNGOP | Novel OP that identifies hydrate based on the number of 5- and 6-membered rings that each water molecule belongs to. Further details provided in Appendix A. |
| CG-$y$ | Total number of cages of type $y$ in the system. Full $5^{12}$, $5^{12}6^2$, and $5^{12}6^4$ cages are identified. Half-cages of types $5^6$ and $5^6 6^1$ are identified. $y = 5^{12}$, $5^{12}6^2$, $5^{12}6^4$, $5^6$ and $5^6 6^1$, full (all full cages), half (all half cages). |
| $\mathrm{FSICA}_x$ | Total number of water molecules in the largest cluster of type $x$, as identified by face-saturated incomplete cage analysis (FSICA). $x$ = FS for face-saturated cages, CC for complete cages, STD for standard cage types, sI, or sII. |

The final value of all OPs is the size of the largest cluster (i.e. first neighbors) of hydrate-like molecules, with exceptions of CG-$y$ and $F_{4\phi}$-$y$.
$F_{4\phi} = \frac{1}{n_d} \sum_i \sum_{j \neq i} \sum_{k \neq i,j} \sum_{l \neq i,j,k} \cos(3\theta_{i,j,k,l})$ where $n_d$ is the number of dihedrals in the cluster.

1. Identify all unique dihedrals ($i$-$j$-$k$-$l$) that can be created from neighboring water molecules. $i$, $j$, $k$, and $l$ must all be unique water molecules. Water molecules $i$ and $j$ must be first neighbors, $j$ and $k$ must be first neighbors, and $k$ and $l$ must be first neighbors. We define first neighbors as all water molecules within $r_{cut}$ = 0.325 nm (distance of the first minima in the sII crystal).

2. Calculate the dihedral angle for all unique dihedrals as the angle between the normal vectors

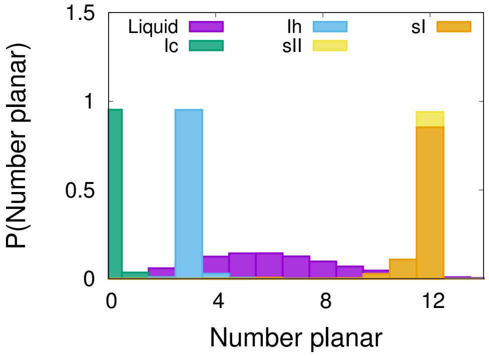

Figure 3.1: Distribution of the number of planar dihedrals that a water molecule participates in for liquid water, sI and sII hydrate, and Ic and Ih for mW at $T = 230$ K.

of the $i$-$j$-$k$ and $j$-$k$-$l$ planes.

3. Mark dihedrals with a dihedral angle below some cutoff (e.g., 30° or 35°) as planar.

4. For each water molecule, increment a counter each time it is part of a central bond (i.e. $j$ or $k$) of a planar dihedral.

5. If a water molecule and at least three of its neighbors are part of 11 or 12 planar dihedrals: (a) tag the water molecule as hydrate-like, and (b) tag all of its first neighbors as hydrate-like.

6. Identify the largest cluster of hydrate-like water molecules, where two water molecules must be first neighbors to belong to the same cluster. No water molecule can belong to more than one cluster.

The distributions of the number of planar dihedrals that a water molecule belongs to for liquid water, structure I hydrate (sI), structure II hydrate (sII), hexagonal ice (Ih), and cubic ice (Ic) are reported in Fig. 3.1. Ice-like, liquid-like, and hydrate-like water molecules can be distinguished based on the number of planar dihedrals that a water molecule participates in.

### 3.2.4 Umbrella sampling

Umbrella sampling was performed to calculate the guest–guest potential of mean force (PMF). Systems consisted of a total of 2000 molecules. The exact number of water and guest molecules were dependent on the guest concentration. Windows were centered every 0.05 nm from

0.2 to 1.5 nm. Spring constants were selected to ensure appropriate overlap between adjacent windows [215]. Systems were equilibrated for 5 ns prior to data collection. Each window was simulated in production for 100 ns with coordinates saved every 0.5 ps. The weighted histogram analysis method as implemented in GROMACS version 4.6.5 was used to construct the final PMFs. Uncertainties were estimated with bootstrapping [216].

### 3.2.5 Straightforward MD nucleation simulations

Twenty straightforward MD simulations were performed with the same system and at the same conditions as the FFS calculations. All trajectories nucleated within 600 ns. Representative snapshots from four trajectories are shown in Fig. 8 of Appendix A.

### 3.2.6 MD Simulation Details

All simulations were performed with the coarse-grained mW model [174]. Parameters for the XL and M guest molecules were taken from Ref. 95. Simulations of hydrate nucleation (both FFS and straightforward MD) were performed in the NpT ensemble at $p = 500$ atm, $T = 230$ K, and 5.6 mol% XL (7555 mW molecules and 445 XL molecules). The conditions of high concentration and large supercooling were selected to allow for large-scale FFS and committor probability analysis, in addition to comparison with nucleation events from straightforward MD simulations. Simulations were performed using the 15 May 2015 version of LAMMPS [188, 189] with a time step of 5 fs. Equilibration simulations used Berendsen temperature [217] and pressure coupling with time constants of 0.5 ps and 1.0 ps, respectively. Nose-Hoover temperature [218] and pressure [219] coupling was used for production simulations, with time constants of 1.0 ps and 5.0 ps for temperature and pressure coupling, respectively. All snapshots were generated with Visual Molecular Dynamics [190].

## 3.3 Results and Discussion

### 3.3.1 Nucleation Rate

FFS calculations converged with a nucleation rate constant of $k_{\mathrm{FFS}} = 1.3 \times 10^{32}$ m$^{-3}$ s$^{-1}$ and resulted in 1101 successful solution–to–hydrate transition paths. The nucleation rate constant was additionally estimated from 20 independent straightforward MD simulations using the technique described by Cox *et al.* [179]. The rate constant was estimated as $k_{\mathrm{MD}} = 6.0 \times 10^{31}$ m$^{-3}$ s$^{-1}$, in good agreement with $k_{\mathrm{FFS}}$. FFS and MD rates for hydrate nucleation have only been previously compared with an order-of-magnitude rate estimate derived from a single straightforward MD nucleation trajectory[104].

### 3.3.2 Committor probability analysis

Accurate committor probabilities ($2\sigma < \pm 0.07$) were calculated for 153 configurations obtained from FFS. Prospective reaction coordinate models created from 33 OPs and linear combinations thereof were evaluated by fitting the models to two different sets of configurations, TP-TPE and TP-NC753. All statistically significant single OP models are ranked in Table 3.2. We note that the exact ordering of top models may change with nucleation conditions. Nonetheless, trends in the rankings of single OP reaction coordinate models offer guidance into the most effective measures to distinguish the hydrate nucleus from the liquid, thus providing insights into the essential features of the nucleation mechanism. For reference, graphs of OP vs. $p_{\mathrm{B,MD}}(\mathbf{x})$ and the linear fits are provided for the top 18 OPs for TP-TPE in Fig. 7 of Appendix A.

The top single OP reaction coordinate models for TP-TPE and TP-NC753 are similar. In both cases, the top models include OPs that are based primarily on water structure rather than guest structure. For example, the DHOP class of OPs identify water molecules that belong to a large number of planar dihedrals – the building blocks of planar 5- and 6-membered rings (5MRs/6MRs) that characterize hydrate structure – as hydrate-like. Meanwhile, the BC class of OPs require that a water molecule be a part of some number of 5MRs and have local tetrahedral order. The appearance of such OPs in the top reaction coordinate models for both TP-TPE and TP-NC753 suggests that water ordering is important in all the transition pathways. Traditional guest-based OPs, such as the original MCG or LSSG OPs, perform poorly as reaction coordinate models for both TP-TPE and TP-NC753. Since MCG has been shown to be a good OP for the nucleation of hydrates of sparingly soluble guest molecules [203], this result suggests the possibility of different nucleation mechanisms for hydrates of soluble and sparingly soluble guests. It is also possible that the differences in the nucleation mechanisms are related to cage occupancy or more subtle aspects of the water–guest and water-mediated guest–guest interactions. Quantification and comparison of hydrate nucleation mechanisms for these different cases remains an open and challenging problem since most studies of clathrate hydrate nucleation are still based upon a limited number of straightforward

Table 3.2: Ranking of single OP reaction coordinate models

| | TP-TPE | | TP-NC753 | |
|---|---|---|---|---|
| Rank | Model | $R^2$ | Model | $R^2$ |
| 1 | DHOP$_{35^\circ}$ | 0.60 | DHOP$_{35^\circ}$ | 0.57 |
| 2 | BC$_{\text{planar}}$ | 0.55 | BC$_{\text{planar}}$ | 0.55 |
| 3 | DHOP$_{30^\circ}$ | 0.52 | CG-$5^6$ | 0.55 |
| 4 | BC-MCG$_1$ | 0.50 | BC$_{56}$ | 0.54 |
| 5 | BC-MCG$_2$ | 0.50 | DHOP$_{30^\circ}$ | 0.54 |
| 6 | RNGOP | 0.47 | CG-half | 0.48 |
| 7 | FSICA$_{\text{FS}}$ | 0.43 | CG-$5^{12}$ | 0.40 |
| 8 | BC-MCG$_3$ | 0.42 | BC-MCG$_1$ | 0.39 |
| 9 | BC | 0.40 | BC | 0.39 |
| 10 | HCOP | 0.40 | HCOP | 0.39 |
| 11 | MCG-6R$_3$ | 0.38 | RNGOP | 0.38 |
| 12 | BC$_{56}$ | 0.38 | CG-full | 0.37 |
| 13 | MCG-6R$_2$ | 0.36 | BC-MCG$_2$ | 0.27 |
| 14 | FSICA$_{\text{CC}}$ | 0.30 | $F_{4\phi}$-BC | 0.27 |
| 15 | MCG$_3$ | 0.28 | FSICA$_{\text{CC}}$ | 0.26 |
| 16 | MCG$_2$ | 0.28 | $F_{4\phi}$-BC$_{\text{planar}}$ | 0.24 |
| 17 | CG-$5^6$ | 0.26 | BC-MCG$_3$ | 0.21 |
| 18 | LSSGOP | 0.26 | FSICA$_{\text{STD}}$ | 0.19 |
| 19 | CG-full | 0.24 | FSICA$_{\text{sII}}$ | 0.16 |
| 20 | CG-half | 0.23 | FSICA$_{\text{sI}}$ | 0.14 |
| 21 | FSICA$_{\text{STD}}$ | 0.14 | CG-$5^6 6^1$ | 0.14 |
| 22 | FSICA$_{\text{sII}}$ | 0.12 | $F_{4\phi}$-BC$_{56}$ | 0.14 |
| 23 | CG-$5^{12}$ | 0.12 | FSICA$_{\text{FS}}$ | 0.12 |
| 24 | – | – | CG-$5^{12}6^2$ | 0.10 |

Not significant at 99% confidence for TP-TPE: CG-$5^{12}6^4$, MCG-6R$_1$, CG-$5^6 6^1$, $F_{4\phi}$-BC, CG-$5^{12}6^2$, $F_{4\phi}$-BC$_{\text{planar}}$, MCG$_1$, $F_{4\phi}$-BC$_{56}$, CG-$5^{12}6^3$, FSICA$_{\text{sI}}$.
Not significant at 99% confidence for TP-NC753: LSSGOP, MCG-6R$_1$, CG-$5^{12}6^4$, MCG-6R$_3$, MCG$_2$, MCG$_1$, CG-$5^{12}6^3$, MCG$_3$, MCG-6R$_2$.

Table 3.3: Top ranked two-OP reaction coordinate models

| Data | Model | $R^2$ | $\Delta$BIC |
|---|---|---|---|
| TP-TPE | $0.011\times$DHOP$_{35^\circ}$ $-$ $0.018\times$CG-$5^6 6^1$ + 0.06 | 0.64 | $-3.1$ |
| TP-NC753 | $0.0039\times$BC$_{\text{planar}}$ + $0.030\times$CG-$5^6$ $-$ 0.55 | 0.69 | $-16.4$ |

MD nucleation trajectories and offer no means for rigorous quantification of the transition state or nucleation mechanism.

OPs based on the number of complete cages in the system perform poorly for TP-TPE. Interestingly, we find that cage-based OPs perform better for TP-NC753. Though the half-cage order parameter (HCOP) performs moderately well for both TP-TPE and TP-NC753, water-based OPs which do not even require complete half-cages (i.e. DHOP, BC) perform better. We suspect that this is because it only takes a single water molecule moving in or out of place to add or remove a half or full cage. Furthermore, HCOP does not identify several less-common complete cage types and incomplete cage-like structures which have been observed during the early stages of hydrate nucleation [204]. DHOP and BC based OPs capture structural changes in water but rely on less specific criteria than the cage-based OPs evaluated in this work. Thus, they likely are more adept at capturing a broader range of cage-like entities and hence the amorphous hydrate structure that commonly appears during hydrate nucleation. By amorphous hydrate structure, we mean a solid-like structure composed of complete and incomplete polyhedral cages but lacking any distinct long-range crystalline order. Though cage-based OPs may provide an accurate estimate of the transition state at conditions of lower driving force, these OPs will struggle to reveal the microscopic details of the earliest stages of hydrate nucleation, which ultimately involve the formation of the first few water cages and consist of $< 100$ water molecules.

Due to the fact that the TP-NC753 configurations belong to transition pathways that are much less sampled by successful solution–hydrate trajectories, TP-NC753 should be representative of higher energy transition pathways. Though $\mathrm{DHOP}_{35^\circ}$ and $\mathrm{BC}_{\mathrm{planar}}$ are the top two single OP models for both sets of configurations, the predicted critical cluster size (i.e. the $p_{\mathrm{B}} = 0.5$ isocommittor surface) is ∼1.5 times larger for TP-NC753 than for TP-TPE. Different critical cluster sizes for TP-TPE and TP-NC753 indicates that TP-NC753 indeed represents a subset of the TPE. This would support the hypothesis that hydrate nucleation occurs through a relatively broad reaction tube, consisting of a variety of different transition pathways [108, 106]. Interestingly, recent work by Kusalik has suggested that the free energy landscape for hydrate nucleation is funnel-shaped, analogous to the free energy landscape for protein folding [220]. This model may help explain the wide variety of amorphous hydrate structures observed in this work as well as most other simulation studies of hydrate nucleation.

Two-OP models are reported for TP-NC753 and TP-TPE in Table 3.3. The improvement compared with the best single-OP model is reported with ΔBIC. ΔBIC of 2–6 is generally considered a notable improvement, 6–10 is strong improvement, and $> 10$ is very strong improvement[221]. The two-OP model only yields slight improvement over single-OP models for TP-TPE but much stronger improvement for TP-NC753.

### 3.3.3 Transition state characterization

Why are water-based OPs generally better at approximating the reaction coordinate than guest-based OPs? To answer this question we first investigated the nature of the TS. The two-parameter reaction coordinate model reported in Table 3.3 for TP-TPE was used to estimate the committor probabilities for the 1573 configurations from FFS which belong to successful transition paths. Configurations with $0.05 < p_{\mathrm{B,est}} < 0.3$, $0.45 < p_{\mathrm{B,est}} < 0.55$, and $0.7 < p_{\mathrm{B,est}} < 0.95$ are considered before, at, and after the TS, respectively. Each group consists of approximately 250 configurations. We note that this approach does not require explicitly calculating the committor probability of all 1573 configurations, but only the OP values, saving significant computational effort. The configurations were visualized with a water-based OP, $\mathrm{DHOP}_{35^\circ}$, and a guest-based OP, MCG-$6\mathrm{R}_2$. We chose to visualize the trajectories with $\mathrm{DHOP}_{35^\circ}$ because it provides the best estimate of the reaction coordinate of any single OP model for TP-TPE. Visualization of configurations with a top-ranked OP ensures that the representations of the hydrate nucleus are close to the true hydrate nucleus. In other words, the evolution of the hydrate nucleus defined by top-ranked OPs is likely relevant to the mechanism of hydrate nucleation. Visualization of the largest $\mathrm{DHOP}_{35^\circ}$ cluster and vicinal molecules suggested that the six-membered water rings and the associated guest pairs appear to play an important role in the nucleation mechanism. Therefore, we also examined MCG-$6\mathrm{R}_2$, which identifies the largest cluster of guest molecules in hydrate-like configurations. To be considered hydrate-like by MCG-$6\mathrm{R}_2$, a guest molecule must belong to two solvent-separated guest pairs, where the guest molecules in the solvent-separated pair must be separated by a six-membered water ring.

Representative snapshots before, at, and after the TS are shown in Fig. 3.2(a)-(c). Before the TS (Fig. 3.2(a)), the largest cluster of water and guest molecules according to $\mathrm{DHOP}_{35^\circ}$ and MCG-$6\mathrm{R}_2$, respectively, rarely overlap. The water molecules in the hydrate nuclei form small cup-like structures consisting primarily of 5MRs. These structures are seemingly at the intersection of two or three partially formed half $5^{12}$ cages.

Configurations at the TS begin to contain structural motifs found in the sII crystal. Addi-

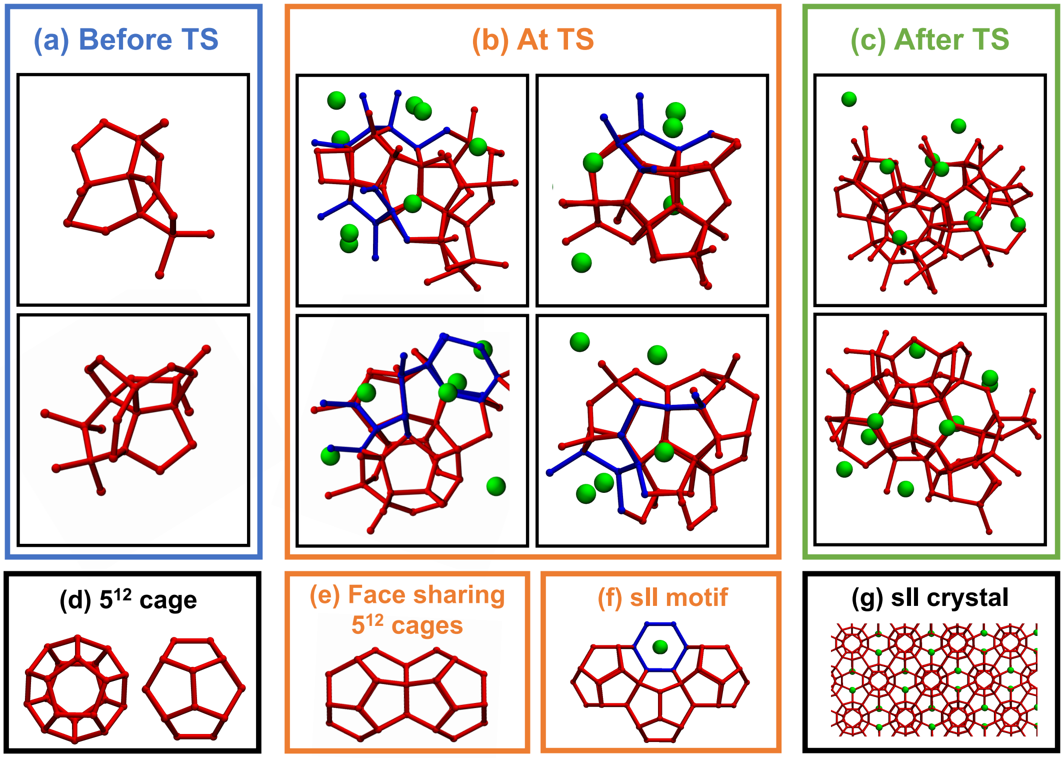

Figure 3.2: Snapshots of representative configurations (a) before the TS ($0.05 < p_{\mathrm{B,est}} < 0.3$), (b) at the TS ($0.45 < p_{\mathrm{B,est}} < 0.55$), and (c) after the TS ($0.7 < p_{\mathrm{B,est}} < 0.95$). Water molecules belonging to the largest cluster of hydrate-like water molecules as identified by $\mathrm{DHOP}_{35^\circ}$ are shown as red spheres and connected by bonds. Guest molecules which belong to the largest cluster of MCG-6$\mathrm{R}_2$ guest molecules are shown as green spheres. Water molecules which are additionally part of a partial or complete 6MR are highlighted with blue spheres and bonds in Panel (b). (d) Two different views of a $5^{12}$ cage. (e) Face-sharing $5^{12}$ cages, and (f) a motif from the sII crystal. Motifs (e) and (f) are observed near the TS. (g) sII crystal.

tionally, the DHOP$_{35^\circ}$ cluster of water molecules and MCG-6R$_2$ cluster of guest molecules (green spheres) spatially overlap. The nuclei appear to form at the junction of face-sharing $5^{12}$ cages. Snapshots of a complete $5^{12}$ cage and two face-sharing $5^{12}$ cages are provided in Fig. 3.2(d) and (e), respectively. Two or more nearly complete face-sharing $5^{12}$ cages are observed in many configurations at the TS. Most configurations also contain at least one partially or completely formed planar 6MR (blue). The 6MRs build from face-sharing $5^{12}$ structures which anchor 3 to 4 water molecules in the 6MR. Guest molecules are positioned on either side of the partial or complete 6MR. The resulting structure forms a motif found in the sII crystal (Fig. 3.2(f)). After the TS, additional structural motifs of the sII crystal appear. Incomplete $5^{12}6^n$ cages form around some guest molecules, additional 6MRs form, and guest molecules near the initial $5^{12}$ cages arrange themselves in a tetrahedrally coordinated network.

We further investigate the transition state using FSICA [204] to identify the largest cluster of complete cages in configurations before and at the transition state. FSICA is capable of identifying all traditional and non-traditional clathrate cages composed of three to six membered water rings. Approximately 60% of the configurations before the TS contained unoccupied, face-sharing $5^{12}$ cages. ∼20% of configurations had no complete cages, and the remaining ∼20% had an occupied cage with no face-sharing $5^{12}$ motifs. At the TS, ∼80% of configurations contain two or more unoccupied, face-sharing $5^{12}$ cages. Sometimes, there is additionally an occupied cage (with 1 or more six-membered rings) that is anchored by the face-sharing $5^{12}$ cages. Furthermore, the ∼20% of configurations at the TS without the face-sharing $5^{12}$ motifs are larger clusters of cages, suggesting that the face-sharing $5^{12}$ motif creates a more stable nuclei. FSICA also confirms that the face-sharing $5^{12}$ motifs first observed with the top single OP, DHOP$_{35^\circ}$, generally form before other complete cages (traditional or non-traditional) and are indeed a key feature of the TS. We are performing ongoing work to further analyze the stability of various hydrate nuclei.

The 20 nucleation trajectories from straightforward MD support the nucleation mechanism observed from FFS. The sII motif shown in Fig. 3.2(f) was frequently observed, and once two or more face-sharing $5^{12}$ cages formed, hydrate often grew. Snapshots from MD trajectories are shown in Fig. 8 of Appendix A. Other nucleation pathways were observed in a few trajectories. In instances where partial cages with 6MRs formed first, the resulting nuclei either dissolved, or took longer to cross the transition region, appeared frustrated in their growth, and resulted in visually more amorphous hydrates. FSICA on straightforward MD nucleation trajectories also commonly revealed the presence of empty, face-sharing $5^{12}$ cages and sII motifs just prior to rapid growth of the hydrate nucleus. Some guest molecules occupied non-traditional cages during the early stages of nucleation, but these occupied cages were most often transient and dissolved quickly.

We hypothesize that partial or complete face-sharing $5^{12}$ cages at the TS provide structural anchoring that enables the formation of planar 6MRs and leads to nucleation. Since the XL guest molecules only occupy $5^{12}6^4$ cages of the sII hydrate, they are unable to be incorporated into the hydrate structure until 6MRs form. It thus seems logical that the formation of planar 6MRs is a key step in the nucleation of sI and sII hydrate where the guest molecules only occupy large cages ($5^{12}6^2$ or $5^{12}6^4$). The appearance of the first 6MRs in the hydrate structure at the TS and the rapid development of additional such structures after the TS, is consistent with this hypothesis.

### 3.3.4 Water and guest structure around transition state motifs

To evaluate our hypothesis we studied water structure around rigid cages. Inspired by motifs (Fig. 3.2(e) and (f)) commonly observed in straightforward MD and FFS, we performed simulations of one, two, and three empty, face-sharing $5^{12}$ cages surrounded by a 5.6 mol% XL–water solution at 500 atm and 270 K. The cages were treated as rigid bodies, and the simulations were performed at higher temperature to prevent hydrate growth.

The structure of water around the rigid cages was analyzed by identifying all primitive 4–7 MRs present in water. A subset of these rings was identified as planar if the maximum dihedral angle of any four consecutive water molecules in the ring was less than 35°. We selected 35° as the cut-off for planar dihedrals because 35° was found to be the most effective definition of planar dihedrals for the DHOP class of order parameters. $\langle R_w \rangle$ is the number of rings that a water molecule participates in, averaged over the water molecules at a given distance from the rigid cages. Since the rigid cage structures are not spherically symmetric, the distance to the rigid cages was calculated as the distance to the closest vertex of a rigid cage. Fig. 3.3 reports the results, normalized by $\langle R_w \rangle$ of bulk XL solution at the same conditions. The rigid cage structures have a limited effect on $\langle R_w \rangle$ for primitive rings. Water molecules near the rigid cage structures belong to fewer primitive 6MRs and 7MRs compared with bulk XL solution (Fig. 3.3(c)-(d)). There is also a slight increase in the number of primitive 5MRs that each water molecule participates in at distances which correspond to the first and second solvation shells of water around the rigid cage structures (Fig. 3.3(b)). However, these changes are less than 20% change from bulk XL solution and are largely independent of the

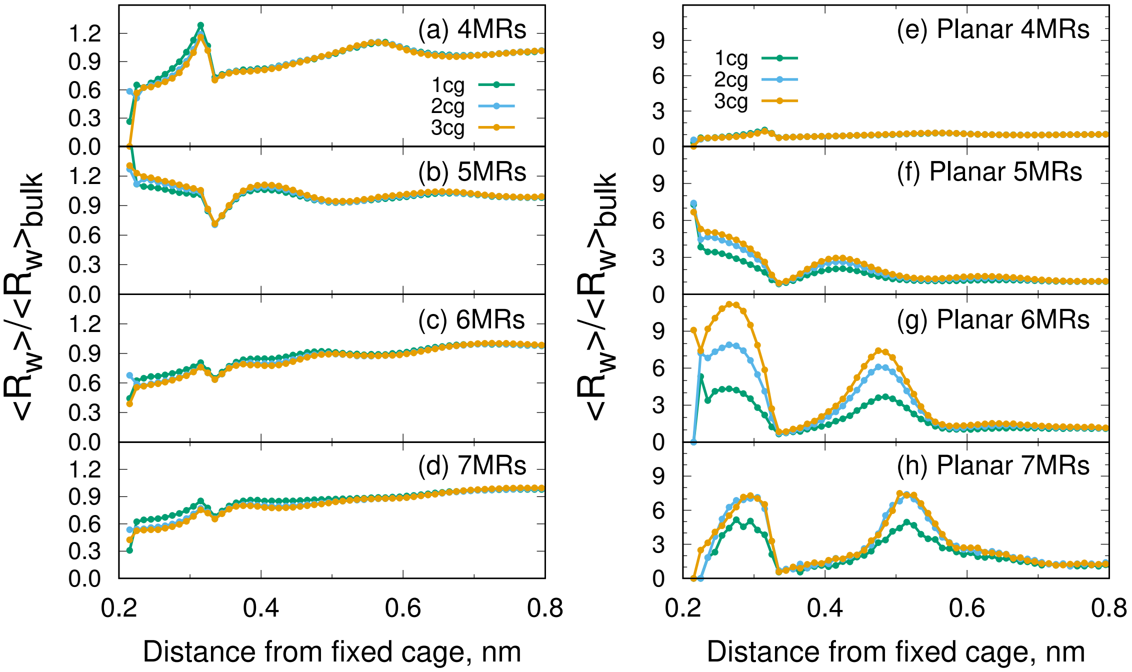

Figure 3.3: Average number of rings, $\langle R_w \rangle$ that a water molecule participates in at a given distance from rigid cages. $\langle R_w \rangle$ is reported for primitive 4–7 MRs in panels (a)–(d) and planar primitive 4–7 MRs in panels (e)–(h). Results are normalized by $\langle R_w \rangle_{\text{bulk}}$.

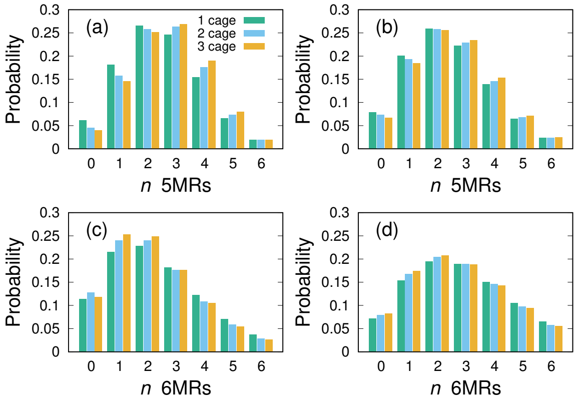

Figure 3.4: Probability that a water molecule within some distance of 1, 2, or 3 rigid $5^{12}$ cages is part of $n$ primitive 5MRs (a,b) or 6MRs (c,d). Panels (a) and (c) are calculated for water molecules that are at a distance from the closest vertex of the fixed cages corresponding to the first peak in the mW–mW RDF in the sII crystal (0.245–0.295 nm). Panels (b) and (d) are calculated for water molecules that are at a distance from the closest vertex of the fixed cages corresponding to the second peak in the mW–mW RDF in the sII crystal (0.394–0.482 nm)

number of rigid, face-sharing $5^{12}$ cages.

The rigid cages have a more pronounced effect on the presence of planar primitive (PP) rings. Water molecules near rigid cages participate in 2–10 times more PP 5–7 MRs compared with bulk XL solution. Water molecules participate in an increased number of PP-5MRs near increasing numbers of face-sharing $5^{12}$ cages. There is a greater increase in $\langle R_w \rangle$ when comparing 2 rigid cages to 1 rigid cage than 3 rigid cages to 2 rigid cages, suggesting that while face-sharing $5^{12}$ cages promote PP-5MRs, the sII motif composed of 3 face-sharing $5^{12}$ cages (Fig. 3.2(f)) and observed at the TS does not further promote the formation of PP-5MRs. The presence of rigid cages has the largest impact on PP-6MRs. Water molecules near 1 rigid cage show an enhancement in PP-6MRs that is similar to the enhancement in PP-5MRs, participating in ∼3 times more PP-6MRs than in bulk XL solution. However, water molecules near 2 and 3 rigid cages participate in ∼7 and ∼10 times more PP-6MRs, respectively (Fig. 3.3(g)). Not only does this show that unoccupied $5^{12}$ cages promote the formation of PP-6MRs, but it also suggests that face-sharing $5^{12}$ cages (such as the sII motif observed at the TS) promote the formation of PP-6MRs even more strongly than single cages. $\langle R_w \rangle$ also shows a notable increase for PP-7MRs near rigid cages. There is no difference in $\langle R_w \rangle$ for PP-7MRs between 2 and 3 rigid cages.

To further investigate the water structure near the rigid cages, the probability that a water molecule was part of some number of primitive 4–7MRs was calculated as a function of distance from the closest vertex of the rigid cages. The largest differences between systems with one, two, and three rigid cages were at distances that correspond to the first and second peaks of the mW–mW radial distribution function (RDF) in the sII crystal. Fig. 3.4 reports the probability of observing a water molecule that belongs to some number, $n$, of primitive 5MRs and 6MRs at those distances. As the number of rigid cages increases, the probability that a water molecule belongs to 0–2 5MRs decreases and the probability that a water molecule belongs to 3–5 5MRs increases. The probability that a water molecule belongs to 1–2 6MRs increases and the probability that a water molecule participates in 3–6 6MRs decreases with an increasing number of face-sharing $5^{12}$ cages. These changes are consistent with the structure of the sI and sII hydrate crystal. The sI crystal contains water molecules that are part of four 5MRs and two 6MRs, five 5MRs and one 6MR, and six 5MRs and zero 6MRs. The sII crystal contains water molecules that are part of five 5MRs and one 6MR, and six 5MRs and zero 6MRs. The changes in the numbers of primitive 5MRs/6MRs that each water molecule participates in occur in addition to the increase in planar primitive water rings near the 1–3 face-sharing $5^{12}$ cages. This suggests a cooperative effect in the development of the correct topological network and the correct spatial arrangements of water molecules to initiate the formation of hydrate structure from the supercooled liquid.

Since guest molecules across 6MRs were observed in the TS structures, we examined their behavior near the rigid, face-sharing $5^{12}$ cages. Previous studies have investigated the effect of cages on surrounding guests. Results indicate that there is an attractive force between the faces of polyhedral cages and guest molecules, resulting in the adsorption of guests to the faces of cages [92, 222]. The attractive force exists whether the polyhedral cage is occupied or not, and distance to the minima in the free energy is different from the distance of the solvent-separated minima of two guests in a liquid solution [92], indicating that the cage face–guest attraction is different from solute–solute interactions in a supercooled liquid solution. We probe this phenomenon further by studying guest structure around not just single cages, but also two and three face-sharing cages, which resemble structures commonly observed at the transition state.

Fig. 3.5 shows the normalized probability of observing a guest molecule that belongs to at least 1, 2, 3, or 4 MCG-6R pairs. The reported probability is normalized by the bulk probability of observing such a structure. Relative to bulk, there is always an increased probability of observing guest molecules in MCG-6R pairs at 0.3–0.5 nm from the rigid cages. Two and three face-sharing $5^{12}$ cages dramatically increase the probability that a guest molecule belongs to at least 2, 3, or 4 MCG-

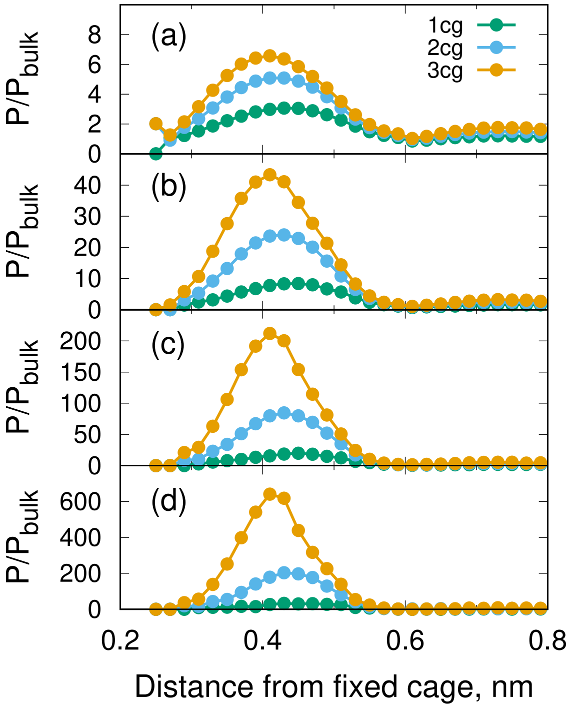

Figure 3.5: Probability that a guest molecule is part of (a) >= 1, (b) >= 2, (c) >= 3, or (d) >= 4 MCG-6R pairs as a function of distance from the nearest vertex of the 1, 2, or 3 fixed cages (green, blue, orange, respectively), normalized by the bulk probability. A MCG-6R pair represents a pair of guest molecules across a 6MR. See Table 3.1 for details.

6R pairs (Fig. 3.5(b)-(d)). Guest molecules near three face-sharing cages are ∼600 times more likely to belong to $>=$ 4 MCG-6R pairs than in bulk solution! A guest molecule that belongs to 4 MCG-6R pairs is tetrahedrally coordinated. Matsumoto has shown that tetrahedrally coordinated methane molecules are significantly more stable than solvent-separated methane pairs [223]. In our work, multiple face sharing $5^{12}$ cages appear to promote tetrahedrally coordinated guest configurations by positioning adsorbed guest molecules in the correct configurations. Furthermore, when guest molecules only occupy the $5^{12}6^{4}$ cages of the sII crystal, they form a tetrahedral network where all guest molecules belong to 4 MCG-6R pairs. Face-sharing $5^{12}$ cages thus promote the guest ordering favorable for sII crystal formation. Our results highlight the symbiotic relationship between water and guest molecules in hydrate nucleation. While guest molecules adsorb to and stabilize water cages [213], the existing water cages facilitate guest molecules to form MCG-6R pairs.

### 3.3.5 Guest–guest interactions in supercooled liquid

Though face-sharing $5^{12}$ cages clearly promote the guest structure requisite for hydrate nucleation, it remains unclear why nucleation requires the formation of this initial water structure. To explain this, we studied the water-mediated XL–XL interactions in bulk solution. We calculated the XL–XL potential of mean force (PMF) with umbrella sampling. For comparison, we also calculated the guest–guest PMF for the M guest molecule. M is a sparingly soluble guest molecule similar to methane or $CO_2$ [95], and has been used in several previous studies of hydrate nucleation [95, 105, 104, 62, 107]. Figure 3.6(a) shows a comparison of the PMFs for the XL and M guest molecules at 500 atm and 240 K. 240 K represents similar supercooling for the thermodynamically preferred crystal structures of XL and M hydrates [95].

The guest–guest PMFs highlight significant differences in the behavior of XL and M in supercooled liquid solution. Solvent-separated configurations of guest molecules are uncommon in XL solution compared with M solution. Furthermore, there is a larger barrier for XL molecules to switch from contact to solvent-separated configurations. The second desolvation barrier is smaller for XL, implying that guest molecules can easily enter and exit solvent-separated configurations. These results likely explain previous observations [95] that amorphous blobs of XL were extremely short-lived compared to amorphous blobs of M. The PMFs suggest that solvent-separated XL configurations are not prevalent or long-lasting enough to promote water structuring. In contrast, M–M PMFs show relatively more stable solvent-separated pairs in the supercooled liquid which could assemble into amorphous blobs.

To further characterize the nature of XL–XL interactions in supercooled liquid solution, PMFs were calculated at multiple XL concentrations and temperatures. Figs. 3.6 (b) and (c) show the concentration dependence of the XL–XL PMF at 240 K and 300 K, respectively. 240 K is close to the conditions of nucleation in this study, and 300 K is approximately 10 K below the melting point of the hydrate. In both cases, the XL–XL PMF shows a concentration dependence. With increasing guest concentration, the contact and solvent-separated minima become shallower and the desolvation barrier height decreases. The effect is so strong that at 5.7 mol% XL there is no recognizable solvent-separated minimum or desolvation barrier. XL guest molecules thus show no preference whatsoever for solvent-separated configurations at these concentrations. For comparison, the concentration dependence of the M–M PMF was calculated at 300 K (Fig. 3.6(d)). M–M interactions have no concentration dependence. The solvent-separated minimum is well-defined and the desolvation barrier remains the same height at all concentrations.

The lack of a distinct solvent-separated minimum at the XL concentration employed in this work emphasizes that stable solvent-separated blobs do not form in the supercooled liquid solution without initial water ordering. In sum, our results indicate that water ordering, rather than guest ordering into amorphous blobs, plays an important role in the early stages of hydrate nucleation for water-soluble guest molecules such as XL. The initial water ordering requisite for hydrate nucleation is analogous to recent findings on the structure of supercooled water, which showed structural heterogeneity on nanometer lengthscales and ice nucleation initiating in pre-ordered regions [224, 31]. Furthermore, we surmise that amorphous blobs of guest molecules observed during the nucleation of sparingly soluble guests are stabilized by water structuring similar to what we observe here.

Fig. 3.6 also emphasizes the generality of our results to conditions beyond those explored in this study. The XL–XL PMFs show similar behavior and concentration dependence at much lower supercoolings, suggesting that our results are applicable at experimentally relevant supercoolings and concentrations. We note that some prospective applications of soluble guests such as THF employ concentrations as high as 5.6 mol% [82].

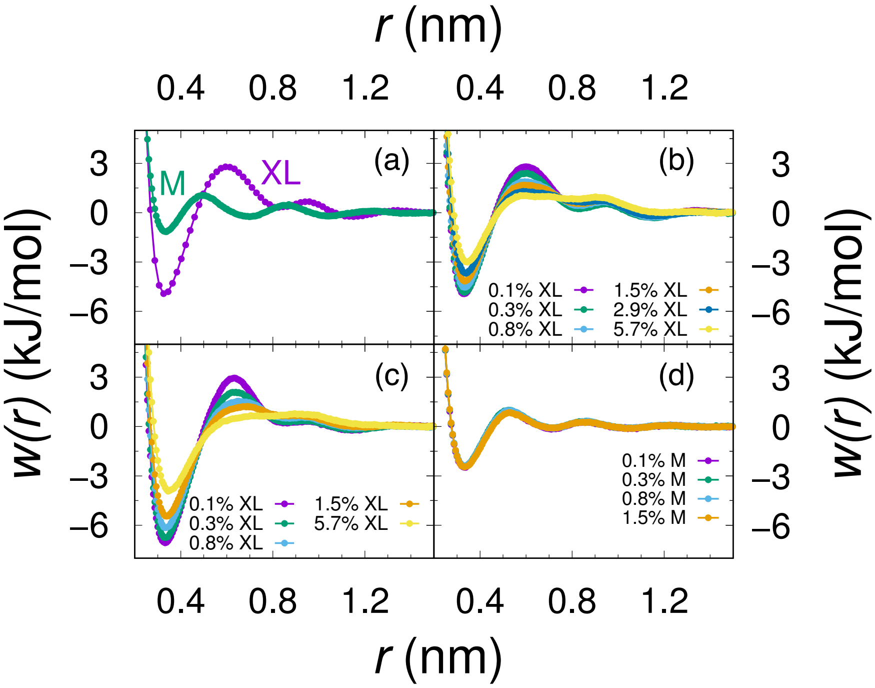

Figure 3.6: Guest–guest PMFs in a supercooled liquid solution at $p$ = 500 atm. (a) Comparison of M–M and XL–XL PMFs at $T$ = 240 K and infinite dilution. (b) Concentration dependence of XL–XL PMFs at $T$ = 240 K. (c) Concentration dependence of XL–XL PMFs at $T$ = 300 K. (d) Concentration dependence of M–M PMFs at $T$ = 300 K. Error bars are approximately the point size.

## 3.4 Conclusions

Despite the relevance of water-soluble guest molecules to applications of hydrates there have been relatively few studies of their nucleation mechanism. To overcome the challenges of sampling nucleation trajectories in simulations, we employ FFS to generate sufficient trajectories for rigorous quantitative analysis of the nucleation mechanism and transition state. We perform an extensive comparison of order parameters for hydrate nucleation and find that order parameters based upon water structure are consistently better approximations of the reaction coordinate compared with order parameters based upon guest structure. In contrast, guest-based order parameters have been successfully employed for hydrates of sparingly soluble guest molecules [105, 203], highlighting potential differences in the nucleation mechanisms for clathrate hydrates with different guest molecules. TS analysis reveals that the empty, face-sharing, partially complete $5^{12}$ cages that characterize the TS dramatically affect the surrounding guest structuring, emphasizing the cooperative relationship between water and guest structure requisite for hydrate nucleation. More generally, water structuring may be important in the nucleation of hydrates of guest molecules that do not fit into $5^{12}$ cages. The comparison of order parameters presented here will facilitate future studies of hydrate nucleation, since order parameters which closely approximate the true reaction coordinate improve the efficiency [225, 144] and correctness of advanced sampling methods [123, 156].

# Chapter 4

# SAFFIRE: A scalable forward flux sampling framework[1]

## 4.1 Introduction

The kinetics relevant to several processes in condensed matter physics such as protein folding, transport through membranes, bubble formation, and crystallization are difficult to study through straightforward molecular dynamics (MD) or Monte Carlo (MC) simulations. This is because the time between occurrences of these rare events can be much longer than the practically accessible timescales of the simulations. At typical MD simulation lengthscales of a few nanometers, observing rare events such as crystallization often requires microseconds-long simulations. These simulations can take several months of computational time for molecular systems like all-atom water models. Given that several hundred rare events are necessary to obtain statistically relevant rate estimates, it is computationally prohibitive to study rare event transitions through straightforward MD (or MC) simulations.

One such process of interest in our research is crystal (e.g., ice) nucleation. Homogeneous and heterogeneous ice nucleation are relevant to atmospheric chemistry and have a significant impact on the climate and weather [227]. The kinetic details such as nucleation rates and mechanisms of ice nucleation, especially in case of heterogeneous ice nucleation, have remained elusive due to several

[1] Material for this chapter adapted from Ref. [226]

difficulties. For example, in experimental studies the nucleation rates calculated are very sensitive to the technique used [228]. Further, the lengthscales (involving few hundreds of water molecules) and the timescales at which nucleation proceeds are hard to probe in experiments. On the other hand, molecular simulations are designed for these length- and time-scales, making them ideally suited for studying ice nucleation. However, since ice nucleation is a rare event, sampling sufficient nucleation events is challenging.

Several techniques [145, 229, 230, 153] have been developed to sample rare events in simulations and are collectively referred to as rare event methods. One such technique is forward flux sampling (FFS) [129, 1]. In FFS, simulations from initial state A to final state B are propagated through non-overlapping interfaces between A and B (see Fig. 4.1). This approach breaks down the low probability A-to-B transition into multiple relatively more probable transitions between intermediate interfaces. Compared with other advanced sampling methods, FFS has several advantages, including applicability to equilibrium and non-equilibrium systems and a comparatively simple and embarrassingly parallel algorithm. The challenge in implementing FFS is that as the difficulty of the problem increases, that is, the probability of the A-to-B transition decreases, FFS becomes extremely computationally demanding. Correspondingly, the amount of data and the number of tasks become difficult to handle with traditional scripting tools. We have experienced this in our studies of heterogeneous ice nucleation.

Motivated by this, we have developed a software framework called **S**calable **A**utomated **F**FS **f**or **I**lluminating **R**are **E**vents (SAFFIRE). Our framework utilizes Cascading [231] and Hadoop [208] to handle the large number of tasks and amount of data required for large scale FFS simulations.[2] In this paper we describe the details of the framework and its scalability, compare our approach to other FFS software, and discuss scientific research enabled by SAFFIRE.

## 4.2 FFS Workflow

The goal of FFS is to divide the extremely low probability A-to-B transition into higher probability transitions between interfaces along the A-to-B pathway (Fig. 4.1). Interfaces are defined by specific values of an order parameter ($\lambda$) that can distinguish between state A and state B. For example, in ice nucleation the number of ice-like water molecules can be used as the order parameter

[2] *FFS simulation* refers to a complete execution of the FFS algorithm, whereas *simulation* refers to a molecular simulation (i.e. MD or MC simulation) which is part of the FFS algorithm.

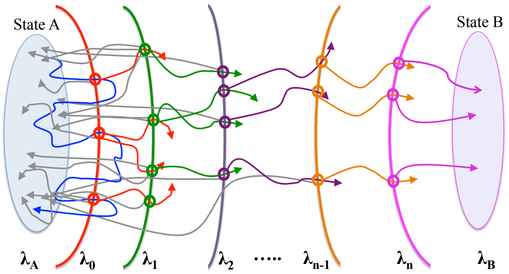

Figure 4.1: Conceptual overview of FFS [1]. The basin simulation is shown as the blue path. Circles represent configurations that are stored at each interface. Trajectories are shown as arrows: trajectories that cross the next interface are colored based on the interface from which they originate; trajectories that return to the basin are shown in gray. $\lambda_i$ represents the i$^{th}$ interface between the A and B basins.

– this value grows as the system transitions from liquid to solid.

FFS starts at the first interface, $\lambda_0$. Configurations for this interface are obtained from simulations in the initial state A ($\lambda<\lambda_{\mathrm{A}}$), also known as the "basin". For each configuration at $\lambda_0$ ($\lambda_i$), several trial simulations are executed using a standard computational code. These simulations are analyzed to identify the next interface, $\lambda_1$ ($\lambda_{i+1}$). Each simulation is then examined to determine whether the simulation trajectory crossed $\lambda_1$ ($\lambda_{i+1}$) or returned to the basin. If the simulation crossed $\lambda_1$ ($\lambda_{i+1}$), the configuration of the system at the instant when the simulation reaches the next interface is added to the set of configurations for $\lambda_1$ ($\lambda_{i+1}$). This set of configurations is used to generate trial simulations from $\lambda_1$ ($\lambda_{i+1}$). This process is continued until the final interface, $\lambda_n$, is reached.

The flow diagram for our implementation of FFS is shown in Figure 4.2. There are three possible outcomes for any trial simulation: (i) the simulation returns to the basin ($\lambda < \lambda_{\mathrm{A}}$) before crossing the next interface ($\lambda > \lambda_{i+1}$), (ii) the simulation crosses the next interface before returning to the basin or (iii) neither (i) or (ii) outcome is obtained. These are referred to as *Fallback*, *Complete*, and *Incomplete*, respectively. As the system moves away from the basin, the simulation time required for simulations to either cross $\lambda_{i+1}$ or enter the basin ($\lambda < \lambda_{\mathrm{A}}$) becomes longer and longer. As a result, two categories of simulations are used. First, "short simulations" are performed, which enable the identification of $\lambda_{i+1}$ and the status of the majority of the trajectories. In the

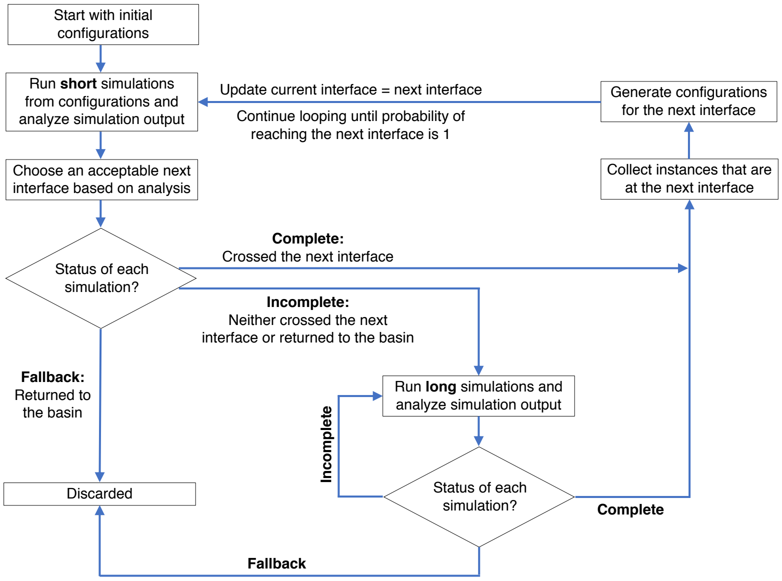
Start with initial configurations
Run **short** simulations from configurations and analyze simulation output
Update current interface = next interface
Continue looping until probability of reaching the next interface is 1
Generate configurations for the next interface
Choose an acceptable next interface based on analysis
Collect instances that are at the next interface
Status of each simulation?
**Complete:** Crossed the next interface
**Incomplete:** Neither crossed the next interface or returned to the basin
**Fallback:** Returned to the basin
Run **long** simulations and analyze simulation output
Incomplete
Status of each simulation?
Complete
Discarded
Fallback

Figure 4.2: Flow diagram for FFS as implemented in SAFFIRE.

case of Incomplete simulations, the simulation neither crosses $\lambda_{i+1}$ nor $\lambda_\mathrm{A}$ in the allotted simulation time. This indicates that the simulation has not run long enough. Incomplete simulations are then extended with "long simulations" until they finish running (become Complete or Fallback). All Complete simulations are then analyzed to generate new configurations for the next interface. The final counts of Fallback and Complete simulations provides the probability of reaching $\lambda_{i+1}$ from $\lambda_i$, $P(\lambda_{i+1}|\lambda_i)$. Once all $N$ interfaces are complete, the product of these probabilities $\prod_{i=0}^{N-1} P(\lambda_{i+1}|\lambda_i)$ is used to estimate the rate of occurrence of the rare event – the transition from initial state A to final state B.

## 4.3 User Requirements

Prior to this work, a framework guiding the FFS workflow was implemented with Bash scripts. It was executed on the campus supercomputer using standard file system support and no additional data infrastructure. Due to I/O bottlenecks, this prototype executed for weeks to complete a small FFS simulation. From this implementation, we learned that for our scientific problems the majority of the simulations require very short execution times (e.g, <5 minutes). However, perhaps millions of simulations are required to successfully complete the FFS simulation. Therefore, it is important to have an infrastructure that can support high throughput computing. Secondly, each simulation produces a modest sized file. The result is that the overall application produces a massive amount of intermediate data from each interface of the FFS simulation. These files are written to the file system and there can be millions of files at any given time. On our campus supercomputing cluster consisting of separate compute and storage nodes, moving, storing, and analyzing this data is a bottleneck for the FFS simulation. In addition, the heavy load leads to instability of the parallel file system. Therefore, we needed to address both the issues of high task throughput and a large number of files.

We identified the following user requirements necessary in a software framework designed to support large scale FFS simulations:

- **Tolerance to single node failure:** The scope and scale of the target application demand substantial computing resources, and the execution times are typically measured in hours or days. Therefore, the framework should be resilient and fault-tolerant to single node and single task failures.

- **Massive data transfer:** FFS requires analysis of aggregated simulation output to determine the status of each simulation and to calculate the probability of advancing to the next interface. The size of the aggregated output requires support for massive data transfer and management capability.

- **Flexible user configuration:** The user needs freedom to choose among different simulation software and analysis tools, depending on the science problem. The framework should also offer flexibility in defining parameters for FFS, such as the number of short simulations, the minimum number of configurations necessary at each interface, etc.

- **Dynamic resource allocation:** Given that the application is being executed in a shared computing environment, it is beneficial for the framework to take advantage of additional resources, or fewer resources, at any time during the execution process in an automated, dynamic, and transparent manner.

- **Usability by a broad community:** The necessary steps for installing, configuring, running, and fine-tuning the framework should be as simple as possible. While not a technical requirement, it is very important in promoting the adoption of the framework by a broad community.

## 4.4 FFS Framework Implementation

SAFFIRE is a comprehensive software framework designed to address the high throughput and data intensive computing challenges presented by FFS. SAFFIRE utilizes the Hadoop infrastructure to control the massive number of individual simulation instances and subsequent output, with the Cascading libraries [231] to manage the overall workflow.

### 4.4.1 Hadoop and Cascading

Hadoop is an open source large scale computing infrastructure that can support the management and processing of a large amount of data based on the principle of data locality [208]. The core Hadoop components include the Hadoop Distributed File System (HDFS) [232] and Hadoop MapReduce (MR) [233]. HDFS is composed of a single centralized management node called the *NameNode*, which maintains all the metadata for the Hadoop infrastructure, along with multiple storage nodes called *DataNodes*, which contain all the data in large block sizes. The MR computation model includes a single central management node called the *ResourceManager*, which is responsible for delegating the specific map and reduce tasks for a submitted MR job to a subset of the NodeManagers located on multiple computation nodes. The DataNodes and NodeManagers exist on the same physical computing system and are connected via communication between the NameNode and the ResourceManager to provide the computation and data locality integration. This is critical to the performance of large-scale data processing. The working mechanisms and performance characteristics of HDFS and MR are well studied [234]. The Hadoop infrastructure comes with features such as scalability, high fault-tolerance, and automated error recovery.

Cascading is a platform that supports the development of complex data-driven applications on the Hadoop infrastructure. Cascading accomplishes this goal by abstracting away the interaction between the developers and the data stored in HDFS. Data dependencies among the different modules or functions of a complex multi-stage application are viewed as data flows, or “pipes”. These data pipes can be manipulated through operations such as filter, merge, split, and redirect. This level of abstraction allows the developer to focus more on the architectural flow of the applications in a plug-and-play manner, rather than the minute interactions with the underlying data.

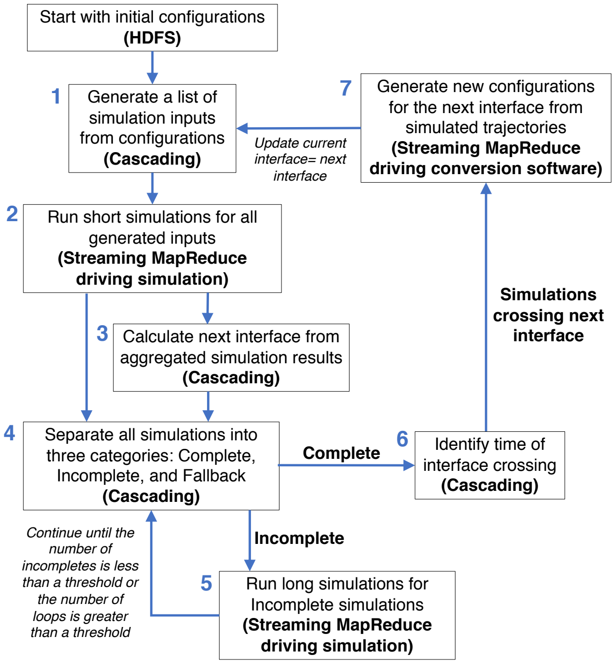
Start with initial configurations
(HDFS)
1
Generate a list of simulation inputs from configurations
(Cascading)
7
Generate new configurations for the next interface from simulated trajectories
(Streaming MapReduce driving conversion software)
Update current interface= next interface
2
Run short simulations for all generated inputs
(Streaming MapReduce driving simulation)
3
Calculate next interface from aggregated simulation results
(Cascading)
Simulations crossing next interface
4
Separate all simulations into three categories: Complete, Incomplete, and Fallback
(Cascading)
Complete
6
Identify time of interface crossing
(Cascading)
Incomplete
Continue until the number of incompletes is less than a threshold or the number of loops is greater than a threshold
5
Run long simulations for Incomplete simulations
(Streaming MapReduce driving simulation)

Figure 4.3: Architectural diagram of SAFFIRE.

### 4.4.2 Implementation Details and Features of SAFFIRE

Key components of SAFFIRE are matched to workflow steps in Figure 4.3. Streaming MR is shown in three components. The first streaming MapReduce (MR) job drives an external program to run the short simulations (box 2). A second streaming MR job drives an external program to run simulations to complete the Incompletes (box 5). A third streaming MR job drives conversion software to generate the set of configurations at the next interface from the simulation trajectories (box 7). Using streaming MR to control external executables enables use of a wide range of simulation engines. We have successfully tested simulation platforms such as GROMACS [235] and LAMMPS [188]. Other software packages for any type of simulation and analysis can easily be integrated with SAFFIRE with no code modifications.

Cascading acts as a flow manager and allows these steps to be executed as a single logical unit while maintaining workload dependencies among the steps. Cascading's ability to manipulate both MR modules and flows of data between these modules enables the addition of data-centric tools into SAFFIRE for the purpose of analyzing intermediate data without impacting the main FFS process. The Cascading workflow implementation is based on previous work in data flow management [209].

The combination of Hadoop and Cascading provides SAFFIRE a number of features that address the application and user requirements outlined in Section 4.2. Through Hadoop, SAFFIRE has the ability to manage the allocated computing resources from user space via the JobTracker's customized scheduler. Hadoop Distributed File System (HDFS) provides large scale data management infrastructure with data locality and data redundancy. The Hadoop platform has mechanisms to automatically support fault-tolerance and error recovery through data replication and job/task re-execution. This renders SAFFIRE a framework that has a high level of fault-tolerance and error recovery capability. We have additionally incorporated a simple user interface into SAFFIRE. Users can modify FFS parameters such as the number of interfaces, the number of simulations per interface, a threshold value used for interface selection, and more. The availability of dynamic Hadoop clusters similar to [236] allows SAFFIRE to be run on any traditional HPC environments. These capabilities improve the usability of SAFFIRE.

The Cascading/Hadoop-based implementation requires only user privileges. No administrative privileges are required to install and run SAFFIRE for the default deployment. This capability has been demonstrated in research and education projects using Hadoop-based environment at scale on the Clemson Palmetto computing cluster [237, 238]. This provides SAFFIRE a high degree of interoperability on different institutional and community platforms such as XSEDE.

## 4.5 Scalability Evaluation

The performance of SAFFIRE is evaluated and discussed in this section, as follows. First, we focus on the scalability of the application with respect to the number of cores and size of the problem using both strong and weak scaling. Second, the behavior of the application (e.g., computation and data transfer) is profiled and characterized under different execution scenarios. Finally, the effects of phases of application performance are characterized under different execution scenarios. Our testbed is part of Clemson University's Palmetto Supercomputer, from which we can provision isolated dynamic clusters to deploy the Cascading/Hadoop environment. Throughout the performance evaluation, the individual compute nodes provisioned for the different experimental clusters are consistently configured with 16-core Intel Xeon E5-2665 CPUs, 64GB of memory, 900GB local HDD, and 300GB local SSD. The system used for the performance analysis was the early stages of homogeneous ice nucleation in the mW water model [174] at 230 K and 1 atm. The system comprised of 4096 water molecules and each simulation was executed for 3 ps (time step = 0.002 ps) of molecular dynamics in LAMMPS (`https://lammps.sandia.gov`) [188]. The order parameter used to quantify the progress of each simulation, $\lambda$, was the size of the largest cluster of ice-like water molecules defined with the procedure from Ref. 50

### 4.5.1 Scalability Analysis

Both strong and weak scaling are considered in the scalability analysis of SAFFIRE. For strong scaling analysis, the problem size (number of simulations per interface[3]) is held constant and the number of cores in the Hadoop cluster used to run SAFFIRE is increased. For weak scaling analysis, the problem size is increased in equal proportion to the increase in the number of cores, so that the amount of work per core remains constant. For both strong and weak scaling analysis, we consider the performance of SAFFIRE for four interfaces of FFS with 10,000 simulations per interface on a 128 core Hadoop cluster as the baseline performance. Strong scaling efficiency, $E_{strong}$ is calculated as:

$$E_{strong} = \frac{t_{128}}{N t_n} \tag{4.1}$$

where $t_{128}$ is the execution time for SAFFIRE on 128 cores (8 nodes), $t_n$ is the execution time for the application on $n$ cores, and $N$ is number of cores in the Hadoop cluster divided by the baseline 128 cores ($n/128$). Weak scaling efficiency, $E_{weak}$ is calculated as:

$$E_{weak} = \frac{t_{128}}{t'_n} \tag{4.2}$$

where, $t_{128}$ is the execution time for the application on 128 cores, and $t'_n$ is the execution time for a problem size $N$ times larger than the baseline problem, executed on $n$ cores, where $n = 128 \times N$.

The scaling performance of SAFFIRE is shown in Figure 4.4 where each test consisted of running four FFS interfaces. Since each FFS interface comprises a similar operation, our results are not expected to change with a larger number of interfaces. The application displays excellent strong scaling performance to more than 600 cores (40 nodes). The execution time drops from nearly 10

[3] For the remainder of the text we use 'number of simulations' in place of 'number of simulations per interface' for brevity (i.e. a 128 core cluster with 10,000 simulations, has 10,000 simulations *per interface*).

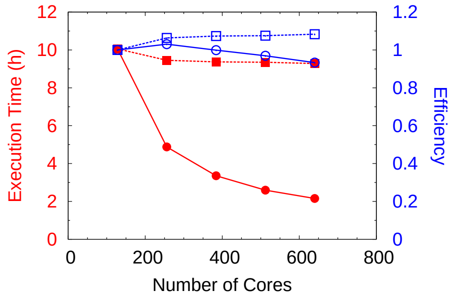

Figure 4.4: Strong and weak scaling of SAFFIRE. Execution time (red, filled markers) is read from the left axis, and scaling efficiency (blue, open markers) from right axis. Strong scaling is shown with circles and solid lines, and weak scaling is shown with squares and dashed lines. The 95% confidence interval is smaller than the symbol size.

hours when running on a Hadoop cluster with 128 cores to just over 2 hours on a 640 core Hadoop cluster. The strong scaling efficiency remains over 90% for all systems tested, however the strong scaling efficiency generally decreases as more cores are added. This may be due to the increased overhead of a larger Hadoop cluster, and increased data transfer times to copy simulation data from HDFS to local scratch and simulation results from local scratch to HDFS. These data transfers are initiated from within the Hadoop Streaming map tasks, and therefore are unable to take advantage of the built-in data-locality offered by Hadoop. As such, when the Hadoop cluster increases in size, the data must be transferred further across the network. Additionally, since HDFS is spread across an increased number of nodes, the likelihood of finding the necessary simulation data already on the node performing the map task decreases. More discussion of data transfer overhead follows later in this section. Larger Hadoop clusters also have the possibility of an increased number of idle nodes if the number of tasks is not divisible by the number of cores available to execute the tasks. We term these “remainder effects” and explore them in detail later in this section.

SAFFIRE weak scaling performance is also shown in Figure 4.4. The overall execution time of the application decreases as the problem size is increased in proportion to the number of cores in the Hadoop cluster. The weak scaling efficiency increases over the range of Hadoop cluster sizes studied, resulting in a weak scaling efficiency that is always greater than or equal to 1. One source for the increase in efficiency is related to how we chose to scale the size of the job in the weak scaling analysis. In our implementation, the head node of the Hadoop cluster does not perform any computation. Therefore, when the number of total cores in the cluster increases, only the number of slave node cores increases – meaning the computational resources available for task execution grows faster than the problem size. While it is possible to take advantage of unused cores on the head node for computational purposes, the amount of memory held by the NameNode and the ResourceManager processes to maintain metadata for the massive amount of data, file counts, and map/reduce tasks makes it impractical to do so. It is also possible to use the number of slave node cores rather than total cores when calculating efficiency. However, we use total cores in the calculation because the head node resources are required to manage the Hadoop cluster, even if they are not being used for scientific computation. The cluster (and therefore application) management overhead decreases in terms of the percentage of total cores with increasing cluster size, and this manifests itself by contributing to the weak scaling performance of the application. Together the strong and weak scaling results highlight the scalability of SAFFIRE.

### 4.5.2 Application Profiling

An analysis of application behavior was performed. The primary goal of SAFFIRE is to efficiently enable the execution of a large number of simulations for FFS. A core is performing useful work when it is running a simulation, analyzing simulation output, or performing a necessary file conversion. All other time is considered application overhead. Realistically speaking, SAFFIRE also manages the simulation output and automates the FFS algorithm. Though these tasks save the end-user time and effort, they are not computation-intensive tasks as compared with the execution and analysis of the simulations. Therefore, as a starting point to evaluate application behavior and framework overhead, we focus on profiling the Hadoop Streaming tasks that are responsible for running the simulations and analysis.

Each Hadoop Streaming map task is executed on one core. Logging capabilities were added to the Hadoop Streaming tasks to record the start and stop times for each simulation, analysis, file conversion, and data transfer. The events were aggregated to create a representation of the number of cores active with each task type across the entire Hadoop cluster at any given instance in time. Logging the start and stop times did not significantly change the overall execution time of SAFFIRE within the 95% confidence interval.

The different types of core activity were grouped into computation (e.g., simulation, analysis, and file conversion), and data transfers (e.g., file transfers between local scratch and HDFS, which

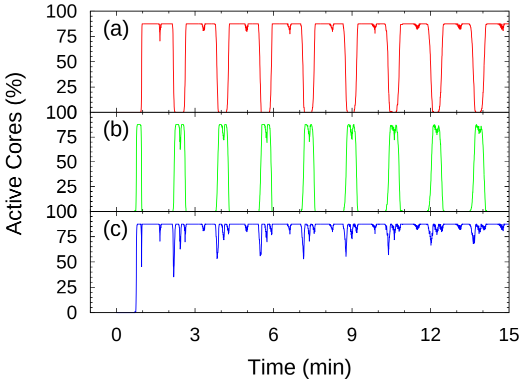

Figure 4.5: Snapshot of application profiling over a 15 minute interval. Percent of cores in the cluster which are (a) performing a computation (simulation or analysis), (b) performing a data transfer, or (c) either performing a computation or data transfer at a given time.

are initiated from within the Hadoop Streaming map tasks). Figure 4.5(a) shows the percentage of all cores performing a computation activity within the 15 minute snapshot. Initially, no cores are active with computation until the ∼1 minute mark, when the first Hadoop Streaming tasks begin. Nearly simultaneously, all the cores on slave nodes are consumed with computation. Note that the percentage of active cores reaches a maximum of about 85% because we report the percentage of active cores with reference to the total size of the Hadoop cluster, not just the number of slave node cores. The 128 core Hadoop cluster shown in Figure 4.5 has up to 112 active cores at one time with the remaining non-active 16 cores of the head node. Evidence of the small time gap between the simulation and analysis appears as a brief decrease in the percentage of active cores between the one and two minute marks. Just past two minutes, none of the cores are involved in computation. In Figure 4.5(b) the percentage of cores involved in data transfers is shown. The cores are involved in a data transfer just before the first computation (Figure 4.5(a)), because the Hadoop Streaming task must copy a configuration file from HDFS to local scratch to initiate the simulation. After the first batch of computation, another batch of data transfers appears as the Hadoop Streaming tasks copy simulation output from local scratch to HDFS. A brief decrease in the data transfer appears (e.g. 2.5 minutes), marking the distinction between the data upload to HDFS from the first batch of simulations, and the data download to local scratch for the second batch of simulations. In Figure 4.5(c), the computation and data activity are combined to report an overall percent of active cores over time.

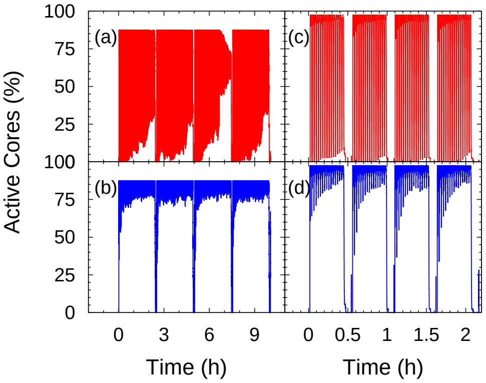

Figure 4.6: Application profiling: Percent of cores in the 128 core cluster with 10,000 simulations which are (a) performing a computation (e.g., simulation, analysis and file conversion), and (b) either performing a computation or data transfer at a given time. Percent of cores in the 640 core cluster with 10,000 simulations which are (c) performing a computation, and (d) either performing a computation or data transfer at a given time.

Several interesting features of SAFFIRE behavior are apparent from Figure 4.5. The simulations are executed across the entire cluster in a batch manner, with all slave node cores actively performing computations and data transfers at nearly the same time. This job submission pattern holds through several batches of Hadoop Streaming map tasks. From the small dips in overall core activity in Figure 4.5(c) there is limited aggregate core downtime between each batch of Hadoop Streaming tasks. The data transfers between local scratch and HDFS also contribute noticeable overhead to SAFFIRE. If a SAFFIRE user had a system that required large file transfers to run a short (in wall clock time) simulation, the data transfer overhead would detrimentally impact the performance of SAFFIRE.

Figure 4.6(a)-(b) shows application profiling across four complete interfaces (i.e. $\lambda_i$ to $\lambda_{i+4}$) of a FFS simulation for the same system profiled in Figure 4.5. For comparison, data is also shown for the case with 640 cores and 10,000 simulations in Figure 4.6(c)-(d). The computation for each FFS interface can be identified as the groupings of core activity (the first of which is from 0 to 2.5 hours and 0 to 0.5 hours for Figure 4.6(a)-(b) and 4.6(c)-(d), respectively), and they are separated by the time interval where the cores are performing neither computation nor data transfers. In these regions, the Cascading code is parsing though the results of the analysis and picking the next interface. Some amount of computation is also performed to generate the new configurations, however it is too short to appear in the plots. The data transfer required for the file conversion appears as the short vertical line in advance of the larger batch of core activity for interfaces 2, 3, and 4 in Figure 4.6(d). The same feature exists in Figure 4.6(b), but is not visible due to the scale of the figure. In Figure 4.6(c)-(d), the batch-like Hadoop Streaming task execution is maintained across each interface. Near the end of the first and last interfaces, the task execution appears to be less synchronized as evidenced by the lines in Figure 4.6(c) not quite reaching zero activity at that part of the execution.

From Figure 4.6(a), it can be seen that the Hadoop Streaming job submission has a "synchronized" nature for the first ~30 minutes for each interface for the system with 128 cores and 10,000 simulations. After about 30 minutes, the Hadoop Streaming jobs do not appear to be synchronized. Some cores are performing computation, while others are performing a data transfer. Neither the white space that appears below the red lines in Figure 4.6(a) as each interface progresses nor above the red lines as interface 3 progresses indicate that the cores are less active overall – just that the Hadoop Streaming task execution does not follow a synchronized pattern after about 30 minutes.

### 4.5.3 Remainder Effects

The appearance of synchronization of the Hadoop Streaming task execution led us to investigate whether making the number of simulations a multiple of the number of slave node cores would decrease the execution time by eliminating "remainder effects", where some cores in the Hadoop cluster are idle while the last partial batch of Hadoop Streaming tasks are completed. Based on the application profiling, we expect that the remainder effects will be most prominent for the system with 640 cores and 10,000 simulations. We also tested the system with 128 cores and 10,000 simulations for comparison. The remainder effects are tested using FFS simulations in which the number of simulations is a multiple of the number of slave node cores (9984 simulations for 624 slave cores, 10080 simulations for 112 slave cores) and then tested using another run in which one additional simulation is added (9985 simulations for 624 slave cores, 10081 simulations for 112 slave cores). This setup tests a worst-case scenario. If the Hadoop Streaming jobs display perfect batch behavior then in the worst case scenario all except one slave node core will be idle when the last simulation is completing. Each test was performed in triplicates to calculate the 95% confidence interval of our results.

Visual inspection of the application profiles (not shown) does not reveal any clear differences between the perfect match and worst case scenario application runs. However, as reported in Table 4.1, there are differences in the execution times. As expected, the remainder effects are the most prominent for the simulation that has the most synchronized-like Hadoop Streaming task execution. For a Hadoop cluster size of 128 cores (112 slave cores), the remainder effects have no significant effect on the execution time. Based on the application profiling seen above (Figure 4.6), this is not surprising because for this setup the synchronous Hadoop Streaming task submission pattern is not present at the end of an interface. For a Hadoop cluster with 640 cores (624 slave cores), the execution time for the system with no remainder simulations is about 200 seconds faster than the worst case scenario. Although this demonstrates that remainder effects can impact the execution time, they represent a small fraction of the overall execution time.

Table 4.1: Execution time and percent of time that the cores were active for systems with possible remainder effects.

| Cores | 128 | 128 |
|---|---|---|
| Simulations | 10080 | 10081 |
| Execution Time (s) | 36390 ± 68 | 36453 ± 140 |
| Computation (%) | 62.0 ± 0.2 | 61.9 ± 0.2 |
| Any Activity (%) | 81.8 ± 0.2 | 81.7 ± 0.1 |

| Cores | 640 | 640 |
|---|---|---|
| Simulations | 9984 | 9985 |
| Execution Time (s) | 7570 ± 73 | 7758 ± 30 |
| Computation (%) | 59.2 ± 0.6 | 57.7 ± 0.3 |
| Any Activity (%) | 78.2 ± 0.8 | 76.3 ± 0.3 |

## 4.6 Related Work

To our knowledge, there are only two other significant efforts in implementing FFS-based simulation techniques, the Flexible Rare Event Sampling Harness System (FRESHS) [239] and Parallel Forward Flux Sampling (PFFS) [240]. The FRESHS architecture includes a FRESHS server and multiple FRESHS clients, all implemented in Python. The server is responsible for accepting parameters for the rare-event simulation, asynchronous communications from clients, and an SQLite database to store data for intermediate interfaces. The server tracks the progress of FFS, while clients are responsible for the simulations and analysis (order parameter calculation).

Similar to SAFFIRE, the goal of FRESHS is to provide a parallelized FFS implementation that allows users to insert various simulation softwares. Beyond the architectural differences, SAFFIRE and FRESHS also differ in the implementation of the FFS algorithm. FRESHS uses the exploring scouts technique [239]. This technique works well when the analysis is run from within the simulation program – however this often requires modification of the simulation software source code. Though FRESHS can be used with separate simulation and analysis codes, it requires extremely short simulations that incur large startup and shutdown overhead. SAFFIRE is specifically designed for separate simulation and analysis codes for maximum user flexibility. The SAFFIRE and FRESHS implementations both have advantages and disadvantages depending on the system and simulation software. Unfortunately their differences make a meaningful performance comparison difficult.

PFFS implements FFS using the C programming language and MPI to support parallelism [240]. In the original design, PFFS is implemented as a single large-scale FFS simulation program. PFFS requires researchers to recompile from source to include custom simulation engines. It also uses individual files on the shared file system to store simulation results which can become difficult to scale as the number of simulations in FFS reaches millions or even billions. The nature of MPI is also a disadvantage of the PFFS implementation as compared to the Hadoop implementation of SAFFIRE. MPI is not typically tolerant to single node or task failures in the parallel computing environment, whereas Hadoop provides fault-tolerance in the case of single node failures. PFFS never matures out of the testing stages, and no software package is available for testing purposes.

## 4.7 SAFFIRE Enabled Science

Our research group is actively using SAFFIRE to study several scientific problems. In addition to the case of heterogeneous ice nucleation described in the introduction, we have also used SAFFIRE to study the nucleation of clathrate hydrates [131] and Lennard–Jones particles. Clathrate hydrates are a crystal composed of water and guest (e.g., methane) molecules. Their formation presents substantial safety hazards in oil and gas transportation. In addition, researchers are exploring hydrates for technological applications in natural gas storage and gas separations. To investigate the mechanism of hydrate nucleation, we used molecular dynamics simulations with FFS comprising 10 interfaces and 10,000–40,000 short simulations per interface. Including the long simulations, SAFFIRE managed over 500,000 individual simulations. The entire calculation required 33 days on a 30-node Hadoop cluster with the same per-node specifications as listed in the scalability evaluation. From the ∼1000 nucleation pathways generated by SAFFIRE, we performed one of the most comprehensive evaluations of the hydrate nucleation mechanism to date [131]. We are currently using SAFFIRE to study the effects of guest solubility on the hydrate nucleation mechanism.

Our group also actively develops new methods to study rare events, and recently published a method called contour FFS (cFFS) [241]. cFFS allows FFS to be performed along multiple order parameters simultaneously. This improves the effectiveness FFS in cases where there are multiple transition tubes and/or the optimal order parameter is not known *a priori*. We are planning to implement cFFS in SAFFIRE.

# Chapter 5

# Contour forward flux sampling: Sampling rare events along multiple collective variables[1]

## 5.1 Introduction

Rare events remain uniquely challenging to study in molecular simulations [242]. These infrequent transitions between long-lived (meta)stable states are characterized by large differences between the timescales of the relevant physics (e.g., molecular vibrations, hydrogen bond lifetimes, etc.) and the time between events (often $\mu$s to s). Exemplars include crystal nucleation [104, 51, 53, 131], ion-pair dissociation in solution [243, 244], conformational changes in biomolecules [119, 156], and chemical reactions [245]. Due to the prevalence and importance of rare events, several advanced sampling methods have been developed [137, 143, 246, 146, 129, 1, 247, 248, 249, 250, 142] to estimate transition rate constants and sample unbiased trajectories connecting the stable states. However, even with increasing computational power some phenomena remain challenging to study and continued method development is required.

We present contour forward flux sampling (cFFS), a novel method to sample rare events

[1] Material for this chapter adapted from Ref. [241]

with multiple collective variables[2] (CVs) simultaneously. Building on forward flux sampling (FFS), cFFS leverages overall trajectory behavior to on the fly determine nonlinear interface placement in multiple CVs. FFS is a rare event sampling method that uses a series of non-overlapping interfaces to drive a system from an initial state $A$ to final state $B$ [129, 1, 153, 251]. Each interface is defined by some value of an order parameter, $\lambda$, which changes monotonically from $A$ to $B$. Straightforward simulation in $A$ is used to estimate the flux, $\Phi_{A0}$, from $A$ to the first interface, $\lambda_0$, and to collect a large number of first-crossing phase points at $\lambda_0$. The designation of a phase point as a first-crossing point indicates that upon following the trajectory backwards in time from the point, one would reach $\lambda_A$ before $\lambda > \lambda_0$. Several trajectories are initiated from each phase point collected at $\lambda_0$ ($\lambda_i$). Stochasticity from the dynamics or velocity perturbation at the start of each simulation ensures trajectory divergence. Trajectories returning to $A$ are discarded, while those reaching the next interface, $\lambda_1$ ($\lambda_{i+1}$), are stored for the next iteration. This procedure is repeated for each interface until the boundary of $B$ is reached, or the probability of advancing to the next interface, $P(\lambda_{i+1}|\lambda_i)$, plateaus to 1. The transition rate constant is calculated as $k_{AB} = \Phi_{A0} \prod_{i=0}^{n-1} P(\lambda_{i+1}|\lambda_i)$ and transition paths from $A$ to $B$ are generated by connecting the partial paths backward from $B$ to $A$. FFS has emerged as a popular choice for studying rare events in simulation because it is applicable to equilibrium and nonequilibrium systems, and implementation is algorithmically straightforward and embarrassingly parallel.

Despite its advantages, FFS has shortcomings. Assuming reasonable definitions for the boundaries of $A$ and $B$, the rate constant and transition path ensemble (TPE) computed with FFS are, in principle, independent of the order parameter used for the calculation [1]. In practice, a poor choice of order parameter is detrimental to the efficiency of FFS [136, 123] and can even lead to incorrect results [154, 123]. This arises when portions of $\lambda_i$ which are important to the transition are sparingly sampled. More formally, imagine some coordinate ($\lambda^{\perp}$) orthogonal to $\lambda$. Challenges arise for FFS when there is poor overlap between the distribution of first-crossing phase points captured at $\lambda_i$, $\rho(\lambda^{\perp}|\lambda_i)$, and the probability of reaching $\lambda_B$ from some point on $\lambda_i$, $P(\lambda_B|\lambda_i;\lambda^{\perp})$.[123] There are two approaches to overcome this issue: (1) increase sampling to collect more phase points at problematic interface(s), or (2) improve the choice of order parameter to increase overlap between the two distributions. The first approach yields more phase points everywhere along an interface,

[2] In this work, a *collective variable* is a quantity that can be calculated from the configuration space coordinates of the system. An *order parameter* is a collective variable that can distinguish between states $A$ and $B$.

but with sufficient sampling the paths spawned from phase points with a higher $P(\lambda_B|\lambda_i;\lambda^{\perp})$ will come to dominate the eventual path ensemble, resulting in the correct rate constant and TPE. Unfortunately, the efficiency of FFS will still be poor. In contrast, the second approach increases the efficiency of FFS, meaning that FFS will converge to the correct rate constant and TPE with less sampling. Unfortunately, optimal order parameters are rarely known a priori. More often, one of the reasons for generating a path ensemble with a method such as FFS is to identify order parameters which best describe the transition.

Since sampling of all interfaces $i > 0$ in FFS depends on the phase points collected at $\lambda_0$, methods have been proposed to optimize placement of, and ensure adequate sampling of $\lambda_0$ [155, 104, 58]. If the situation is not too dire, increasing the length of the basin simulation and collecting more phase points at $\lambda_0$ may provide a sufficient remedy. However, if overlap between $\rho(\lambda^{\perp}|\lambda_0)$ and $P(\lambda_B|\lambda_0;\lambda^{\perp})$ is extremely small, this may be insufficient. Furthermore, the problem is not limited to $\lambda_0$; in principle the distribution of phase points sampled at any $\lambda_i$ could suffer from this problem. Poor overlap between $\rho(\lambda^{\perp}|\lambda_i)$ and $P(\lambda_B|\lambda_i;\lambda^{\perp})$ becomes particularly problematic for systems with multiple transition tubes. There, a poor choice of order parameter may result in some transition tubes becoming (artificially) favored over others. In the extreme, entire transition tubes can be missed by FFS.

A related situation worth mentioning is when $\rho(\lambda^{\perp}|\lambda_0)$ converges extremely slowly [104, 58]. If this is the problem, extending the basin simulations until convergence is achieved will remedy the situation [104]. A greater number of phase points at $\lambda_0$ are not required; just phase points correctly sampled from the converged distribution.

The choice of order parameter strongly affects the overlap between $\rho(\lambda^{\perp}|\lambda_i)$ and $P(\lambda_B|\lambda_i;\lambda^{\perp})$. If the order parameter is the committor function, $P(\lambda_B|\lambda_i;\lambda^{\perp})$ is constant with $\lambda^{\perp}$, thereby assuring good overlap between $\rho(\lambda^{\perp}|\lambda_i)$ and $P(\lambda_B|\lambda_i;\lambda^{\perp})$ [123]. Borrero and Escobedo thus devised a method to optimize the order parameter with a series of FFS simulations [144]. Though the approach yields improvements [156], it is challenging for systems which require extraordinary computational resources for even a single FFS run [53, 131]. Furthermore, some processes are inherently multidimensional [31, 230, 252, 243], and driving the transition along a single CV may not be ideal.

cFFS takes a different approach. We extend FFS to use multiple CVs on the fly. This allows researchers to test multiple CVs simultaneously and improves the chances of capturing important orthogonal coordinates within the set of CVs used to drive the transition. At each interface, cFFS identifies the next interface as a nonlinear combination of specified CVs computed on the fly from the behavior of simulations initiated from the previous interface. In doing so, cFFS also reveals the role of each CV through the entire transition. Only the combination of CVs must separate $A$ and $B$ and so each CV need not monotonically change from $A$ to $B$. If some CV is unimportant, this will be reflected by, but not impede cFFS. These features offer substantial flexibility in CVs that can be used with cFFS. cFFS generates an estimate of the transition rate constant and a collection of $A \rightarrow B$ trajectories belonging to the TPE. We demonstrate cFFS with two CVs, but in principle it can be extended to three or more CVs.

In Sec. 5.2 we explain cFFS. We proceed to demonstrate cFFS on several two-dimensional potential energy surfaces in Sec. 5.3. In Sec. 5.4, we demonstrate cFFS with one position coordinate and one momentum coordinate, and in Sec. 5.5 we test cFFS on a standard higher dimensional test case, a conformational transition in alanine dipeptide. Discussion and closing remarks are provided in Sec. 5.6 and Sec. 5.7, respectively.

## 5.2 Contour forward flux sampling

The central idea of cFFS is to allow the system to naturally evolve along multiple CVs to reveal how different CVs participate in the transition. This is achieved by placing the subsequent interface based on sampling initiated from the current interface. The FFS formalism can still be used to calculate the rate constant and TPE. Interface placement is designed such that the distribution of first-crossing points is uniform along the interface, ensuring that each interface is well-sampled everywhere within the chosen CVs.

The first step of cFFS is to run straightforward basin simulations in $A$ to identify the bounds of $A$ ($\lambda_A$) and the first interface ($\lambda_0$), and to collect phase points at $\lambda_0$. The value of each CV in time, $\boldsymbol{\lambda}(t)$, is calculated, where $\boldsymbol{\lambda} \equiv \{\lambda^I, \lambda^{II}, \ldots, \lambda^N\}$ is the set of CVs. CV space is discretized to create an $N$-D grid. The discretization size is selected such that the system rarely travels more than a single grid site in one time step. The discrete probability distribution, $P(\boldsymbol{\lambda})$, is calculated from the basin simulations. Grid sites exceeding a threshold probability are added to the set of sites describing $A$, $\boldsymbol{s}_A$. Regions of CV space which are not in $\boldsymbol{s}_A$ but completely surrounded by $\boldsymbol{s}_A$ are added to $\boldsymbol{s}_A$. $\lambda_A$ is defined as the boundary between sites in $\boldsymbol{s}_A$ and those that are not. Trajectories exit $A$ when they cross from a grid site in $\boldsymbol{s}_A$ to a grid site not in $\boldsymbol{s}_A$.

Several criteria are used to identify $\boldsymbol{s}_0$, the set defining $\lambda_0$. $\boldsymbol{s}_0$ should: (a) completely contain $\boldsymbol{s}_A$ so that $\lambda_0$ does not overlap with or cross $\lambda_A$, (b) not create regions of CV space completely surrounded by $\boldsymbol{s}_0$, but not included in it, (c) not include sites in $\boldsymbol{s}_B$, the set of sites describing $B$, (d) be selected such that some desired number of phase points can be collected at $\lambda_0$, and (e) be selected such that there is equal flux of trajectories exiting $\boldsymbol{s}_0$ along the entire $\lambda_0$ interface. Criteria (e) is crucial as it ensures that cFFS does not bias the system to sample any one direction more readily than another. Further discussion is provided later. Once $\lambda_A$ and $\lambda_0$ are defined the basin simulations are re-analyzed to calculate $\Phi_{A0}$ and collect phase points at $\lambda_0$.

The remainder of cFFS proceeds as follows. Several trajectories are initiated from each phase point at $\lambda_i$ ($\lambda_i = \lambda_0$ for the first iteration). Trajectories are terminated when they return to $\lambda_A$, or reach a maximum number of steps. The set of sites defining $\lambda_{i+1}$, $\boldsymbol{s}_{i+1}$, is determined from the behavior of trajectories initiated at $\lambda_i$ using analogous criteria to those described for determining $\lambda_0$. Note that $\boldsymbol{s}_{i+1}$ must completely contain $\boldsymbol{s}_i$ to satisfy the effective positive flux formalism [146, 229]. Once $\boldsymbol{s}_{i+1}$ is identified, trajectories are re-analyzed to determine if they cross $\lambda_{i+1}$ (i.e., exit $\boldsymbol{s}_{i+1}$) before returning to $A$. For each trajectory that crosses $\lambda_{i+1}$, the phase point at the time step which the trajectory crosses $\lambda_{i+1}$ is saved. Trajectories which fail to reach $\lambda_{i+1}$ or return to $A$ before the maximum number of steps are extended until they reach $\lambda_{i+1}$ or return to $A$. The probability, $P(\lambda_{i+1}|\lambda_i)$, is calculated from the number of trajectories that reach $\lambda_{i+1}$ before returning to $A$.

Eventually, sites in $\boldsymbol{s}_{i+1}$ will be adjacent to sites in $\boldsymbol{s}_B$. Trajectories initiated from $\lambda_i$ can then reach $\lambda_{i+1}$, return to $A$, or proceed directly to $B$. This indicates the kinetic barrier has been surmounted and thus cFFS is nearly complete. Two probabilities are now calculated; $P(\lambda_{i+1}|\lambda_i)$ and $P(\lambda_B|\lambda_i)$. Our approach is to continue cFFS until $\boldsymbol{s}_{i+1}$ surrounds $\boldsymbol{s}_B$. At this point, $i$ becomes the final interface, $n$. Trajectories initiated from $\lambda_n$ are continued until they reach $\lambda_B$ or return to $\lambda_A$ to close the probabilities for the rate calculation. As with multi-state FFS [253], the transition rate constant is calculated as

$$k_{AB} = \Phi_{A0} \sum_{j=0}^{n} P(\lambda_B|\lambda_j) \prod_{i=0}^{j-1} P(\lambda_{i+1}|\lambda_i). \tag{5.1}$$

The collection of trajectories comprising the TPE is constructed by connecting the partial paths backwards from $B$ to $A$. Note that all trajectories *do not* have equal weight in the TPE. The relative weight of each trajectory is $w = 1/\prod_{i=0}^{j} k_i$, where $j$ is the final interface crossed by a trajectory before reaching $B$ and $k_i$ is the number of trajectories initiated from each configuration at interface $i$.

## 5.3 Demonstration on 2D potential energy surfaces

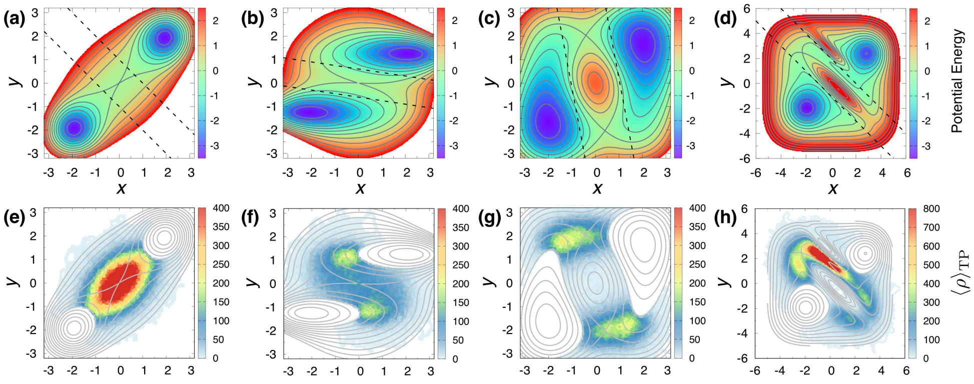

Figure 5.1: *Top panels:* PESs used to test cFFS: (a) PES-1, (b) PES-2, (c) PES-3, and (d) PES-4. Color represents the potential energy. Contour lines are separated by 0.5 units. The region between the dashed lines was used to quantitatively compare $\rho(q|\mathrm{TP})$ between different methods. *Bottom panels*: TPE sampling from SLD at $\beta = 2.5$ on (e) PES-1, (f) PES-2, (g) PES-3, and (h) PES-4.

We demonstrate cFFS with Langevin dynamics of a single particle on four 2D potential energy surfaces (PESs) with different topographical features (see Fig. 5.1(a)–(d)). PES-1 has a single transition tube which follows two monotonically increasing CVs. PES-2 has a single transition tube with hysteresis in the $x$ coordinate. PES-3 and PES-4 both contain two transition tubes; the potential energy barriers are the same for the two tubes on PES-3, and different for the two tubes on PES-4. For each PES, we study $A \to B$ transitions with straightforward Langevin dynamics (SLD), $\mathrm{FFS}_{\mathrm{opt}}$, $\mathrm{FFS}_{\mathrm{x}}$, and cFFS. $\mathrm{FFS}_{\mathrm{opt}}$ denotes FFS performed with the optimal linear combination of $x$ and $y$ (i.e., the order parameter orthogonal to the dividing surface of the PES), and $\mathrm{FFS}_{\mathrm{x}}$ indicates FFS performed with $x$ as the (suboptimal) order parameter. We stress that optimal order parameters are not known a priori for most realistic systems, and therefore FFS is generally performed with suboptimal order parameters. Further details of the PESs, Langevin dynamics, and FFS/cFFS parameters are provided in Appendix B.

Table 5.1: $A \to B$ transition rate constants for four 2D PESs. One standard deviation of the mean is reported in parenthesis.

| | $k_{AB} \times 10^5$ at $\beta = 2.5$ | | | |
|---|---|---|---|---|
| PES | SLD | $\mathrm{FFS_{opt}}$ | $\mathrm{FFS_x}$ | cFFS |
| PES-1 | 2.9 (0.2) | 2.8 (0.3) | 3.1 (0.9) | 2.8 (0.2) |
| PES-2 | 9.1 (0.1) | 7.9 (0.9) | 10.2 (2.5) | 8.8 (0.7) |
| PES-3 | 2.6 (0.3) | 2.4 (0.4) | 2.3 (0.6) | 2.4 (0.1) |
| PES-4 | 1.1 (0.1) | 1.0 (0.1) | 1.1 (0.1) | 1.0 (0.1) |
| | $k_{AB} \times 10^9$ at $\beta = 5.0$ | | | |
| PES-1 | 5.4 (1.2) | 4.4 (0.2) | 3.1 (0.2) | 4.5 (0.6) |
| PES-2 | 23.2 (2.0) | 18.0 (3.5) | 18.3 (1.0) | 21.9 (2.3) |
| PES-3 | 6.4 (1.3) | 2.9 (0.2) | 2.5 (0.2) | 5.4 (0.5) |
| PES-4 | 2.8 (0.9) | 1.9 (0.4) | 0.42 (0.02) | 2.6 (0.1) |

### 5.3.1 Rate constants

$A \to B$ transition rate constants are reported in Table 5.1. Transitions were studied at $\beta = 2.5$ and $\beta = 5.0$ ($\beta = 1/k_BT$). The higher temperature ($\beta = 2.5$) enables rigorous comparison of TPE sampling with SLD, whereas the lower temperature ($\beta = 5.0$) provides a test at more challenging conditions. SLD rate constants are unbiased estimates. $\mathrm{FFS_x}$ provides accurate estimates of the rate constants at $\beta = 2.5$, but at $\beta = 5.0$ $\mathrm{FFS_x}$ underestimates the rate constants. This suggests that suboptimal order parameters perform worse as the barrier becomes larger relative to $k_BT$. We explain the breakdown of $\mathrm{FFS_x}$ by examining the TPE sampling below. $\mathrm{FFS_{opt}}$ and cFFS perform better. Rate constants from $\mathrm{FFS_{opt}}$ and cFFS both agree nicely with SLD at $\beta = 2.5$. At $\beta = 5.0$, $\mathrm{FFS_{opt}}$ underestimates rate constants for PES-2 and PES-3. In contrast, cFFS provides correct estimates of the rate constants for all four PESs at $\beta = 5.0$.

### 5.3.2 Transition path ensemble sampling

Though attaining the correct $A \to B$ rate constant is a crucial test of cFFS, it is also important that cFFS correctly samples the TPE. TPE sampling is calculated as $\langle \rho \rangle_{\mathrm{TP}} = \langle n_{\mathrm{visits}}/l^2 \rangle_{\mathrm{TP}}$, where $\langle ... \rangle_{\mathrm{TP}}$ indicates an ensemble average over all transition paths, and $n_{\mathrm{visits}}$ is the number of times a transition path visited each $l \times l$ region of space. For reference, TPE sampling from SLD at $\beta = 2.5$ is shown in the bottom panels of Fig. 5.1.

Fig. 5.2 summarizes the behavior of $\mathrm{FFS_{opt}}$, $\mathrm{FFS_x}$, and cFFS on PES-1–PES-4 at $\beta = 5.0$. All methods result in qualitatively similar sampling for PES-1. The other surfaces proved more

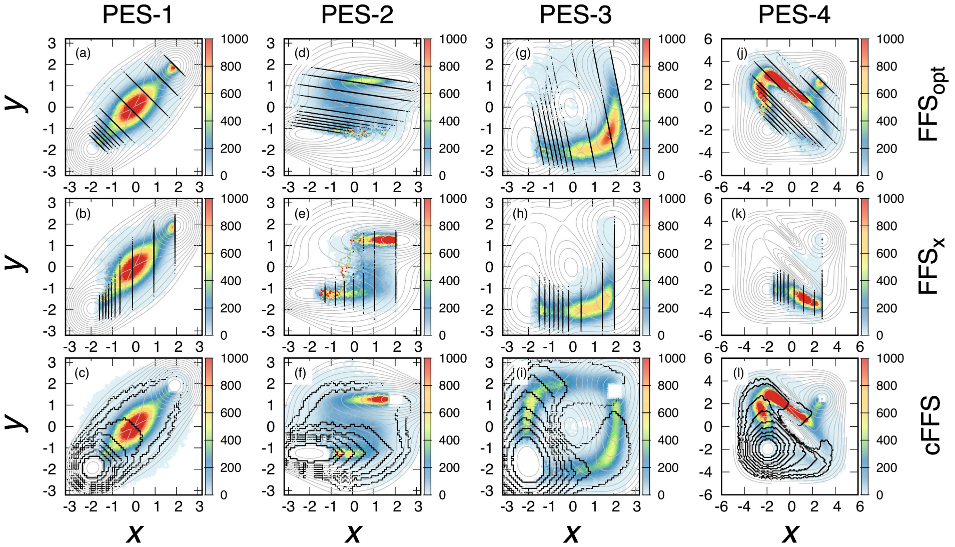

Figure 5.2: Comparison of interface placement and TPE sampling generated with FFS$_{\mathrm{opt}}$, FFS$_{\mathrm{x}}$, and cFFS on PES-1 – PES-4 at $\beta = 5.0$. PES contours are shown as gray lines. Configurations collected at each interface are shown with black points. TPE sampling represented by the heat map.

challenging for FFS$_{\mathrm{x}}$ and FFS$_{\mathrm{opt}}$. In contrast, cFFS results in the qualitatively correct sampling for all four PESs. On PES-2, the hysteresis provides a challenge for FFS$_{\mathrm{x}}$. Unlike FFS$_{\mathrm{opt}}$ and cFFS, FFS$_{\mathrm{x}}$ undersamples the $x < 0$ portion of the transition tube. On PES-3 and PES-4, the failure of FFS$_{\mathrm{x}}$ is even more stark; FFS$_{\mathrm{x}}$ only samples one of the two transition tubes. Even FFS$_{\mathrm{opt}}$ fails to sample both transition tubes equally on PES-3. PES-3 and PES-4 have two distinct transition tubes, and the minimum energy paths change direction from $A$ to $B$. On PES-3, both transition tubes have the same potential energy barrier. However, one transition tube approaches the transition state from $A$ with a gentler slope. Results from SLD at $\beta = 2.5$ in Fig. 5.1(g) indicate that both transitions should be equally traveled. cFFS reproduces this behavior at both $\beta = 2.5$ (Fig. 1 of Appendix B) and the more challenging $\beta = 5.0$ (Fig. 5.2(i)). At $\beta = 5.0$, FFS$_{\mathrm{x}}$ only samples a single transition tube (Fig. 5.2(h)). Even FFS$_{\mathrm{opt}}$ struggles to sample both transition tubes equally on PES-3 (Fig. 5.2(g)). The behavior of FFS$_{\mathrm{opt}}$ and FFS$_{\mathrm{x}}$ on PES-3 can be explained by the framework put forth in the introduction. In both cases, it is apparent that $\rho(\lambda^{\perp}|\lambda_0)$ sampled during the basin simulations only has good overlap with $P(\lambda_B|\lambda_0; \lambda^{\perp})$ for one of the two transition tubes. The result is that FFS oversamples the tube with greater overlap, at the expense of the other transition tube. FFS sensitivity to the choice of order parameter on PES-3 is further demonstrated in Fig. 2 of Appendix B. Though FFS will converge to the correct TPE in the limit of infinite sampling, as a practical

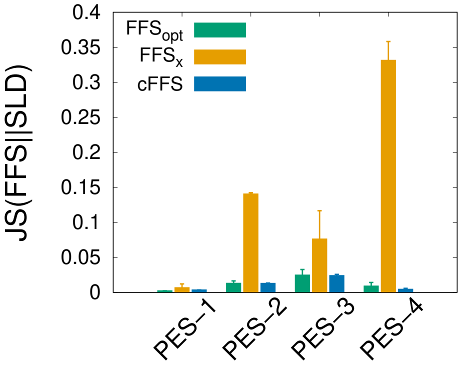

Figure 5.3: Jensen-Shannon divergence between $\rho(q|\mathrm{TP})$ calculated with SLD and $\mathrm{FFS_{opt}}$, $\mathrm{FFS_x}$, and cFFS at $\beta = 2.5$. A value of zero indicates identical probability distributions, while a value of 1.0 indicates completely non-overlapping distributions. Inset focuses on 0.0–0.03 y-axis bounds. Error bars represent one standard deviation on the mean of three independent trials.

matter FFS can lead to incorrect results. cFFS again performs well on PES-4, illustrating that cFFS is able to navigate a tortuous transition landscape with two transition tubes and unequal potential energy barriers.

Near the dividing surface (see Fig. 5.1) we quantitatively compare the TPE density of states, $\rho(q|\mathrm{TP})$, from SLD with that from $\mathrm{FFS_{opt}}$, $\mathrm{FFS_x}$, and cFFS using the Jensen-Shannon divergence [254]. We restrict our comparison to $\beta = 2.5$, where a large number of transitions can be generated with SLD, hence providing a robust reference. The results shown in Fig. 5.3 confirm qualitative conclusions from Fig. 5.2 ($\beta = 5.0$) and Fig. 1 of Appendix B ($\beta = 2.5$). At $\beta = 2.5$, $\mathrm{FFS_{opt}}$ and cFFS perform similarly. For the simplest case (PES-1), $\mathrm{FFS_x}$ performs nearly as well as $\mathrm{FFS_{opt}}$ and cFFS. However, for the more complex surfaces, including the surface with hysteresis (PES-2), and surfaces with two transition tubes (PES-3, PES-4), $\mathrm{FFS_x}$ performs notably worse.

### 5.3.3 cFFS interface placement

Fig. 5.2 also demonstrates cFFS interface placement. Interfaces are spaced further apart in directions that trajectories more readily advance and closer together in directions that trajectories struggle to advance. For these low-dimensional systems, interface locations adhere closely to the contours of the PESs. We strongly emphasize that no knowledge of the PES is employed; cFFS places interface $\lambda_{i+1}$ from the progress of trajectories initiated from $\lambda_i$ alone.

If not done properly, performing FFS with multiple CVs simultaneously can bias the system to over-sample or under-sample regions of CV space. The amount of work performed by FFS is related to interface spacing (i.e., $\lambda_{i+1} - \lambda_i$), slope of the free energy landscape between $\lambda_i$ and $\lambda_{i+1}$, and the number of trajectories initiated from $\lambda_i$. If the slope of the free energy landscape between two interfaces becomes steeper, $\lambda_{i+1}$ is moved closer to $\lambda_i$ or the number of trajectories initiated from $\lambda_i$ is increased. Multiple CVs introduces a new prospect; that unequal amounts of work are inserted along different CVs, biasing the system to over-sample in the direction that more work is inserted.

We introduced a condition of constant flux along an interface in cFFS interface placement to address this problem. The force exerted by the underlying free energy surface is proportional to $-dn_{\text{cross}}/d\lambda$, where $n_{\text{cross}}$ is the number of trajectories crossing an interface placed at some value of $\lambda$. If $n_{\text{cross}}$ changes more quickly with changing $\lambda$, then the underlying surface must have a steeper slope. Applying the differential definition of work, $dW = Fd\lambda$, and thus $dW \propto dn_{\text{cross}}$ and $W \propto n_{\text{cross}}$. Constant flux along the interface requires that all small sections of $\lambda_{i+1}$ have approximately the same number of trajectories crossing them. This condition ensures that equal work is inserted everywhere along the interface (i.e., in all directions) and results in $\lambda_{i+1}$ closer to $\lambda_i$ in directions trajectories struggle to advance and further from $\lambda_i$ in directions trajectories readily advance. The fact that cFFS is able to reproduce the correct TPE symmetry for PES-3 and PES-4 provides strong evidence that the constant flux along the interface condition is correct.

In complex systems, the optimal order parameter is often expected to be a combination (linear or nonlinear) of multiple (suboptimal) order parameters. This combination is generally nonintuitive and difficult to predict. As such, most applications of FFS use a suboptimal order parameter (e.g., $\text{FFS}_\text{x}$). On the four PESs, cFFS successfully produces correct TPE sampling without knowing how $x$ and $y$ should be combined. Though $x$ and $y$ are part of the optimal order parameter, independently, $x$ and $y$ are suboptimal order parameters. This suggests that cFFS can outperform FFS when multiple suboptimal order parameters are known, but the optimal order parameter remains unknown. In addition, nonlinear combinations of CVs have increased degeneracy compared with linear combinations of CVs in creating reaction coordinates (i.e., optimal order parameters) [255]. Since cFFS interfaces are arbitrarily complex combinations of the specified CVs, there may be substantial flexibility in selecting good CVs for cFFS. A variety of approaches have been proposed for identifying important CVs for rare event transitions [256, 210, 127, 257, 258, 144, 211, 135]. For example, recent work suggests that important CVs can be identified from local fluctuations in the (meta)stable basins [259]. We envision using such approaches to identify key CVs for cFFS.

## 5.4 cFFS with a momentum coordinate

FFS is most often applied in the diffusive limit and the CVs used as FFS order parameters are generally only functions of the atomic coordinates. In this section, we demonstrate cFFS on a simple analytical potential where momentum plays a key role during the transition. A previous study shows that FFS fails and under-predicts the transition rate constant when using a position-based order parameter alone [123].

Ref. 123 tested several path sampling methods for a transition on a simple 1D analytical potential described by $V(r) = r^4 - 2r^2$. Of the tested methods, replica exchange transition interface sampling (RETIS) and partial path transition interface sampling (PPTIS) provided the best estimates to the reference effective positive flux (EPF) rate ($k_{AB}^{\text{EPF}} = 2.4 \pm 0.1 \times 10^{-7}$, $k_{AB}^{\text{RETIS}} = 2.8 \pm 0.7 \times 10^{-7}$, $k_{AB}^{\text{PPTIS}} = 2.7 \pm 0.6 \times 10^{-7}$). FFS performed worst, underestimating the rate constant by 1–2 orders of magnitude depending on the length of the basin simulation. With a basin simulation of 4 million steps, FFS produced a rate constant of $k_{AB}^{\text{FFS-short}} = 2.2 \pm 0.2 \times 10^{-9}$. When the basin simulation was extended to 10 million steps, $k_{AB}^{\text{FFS-long}} = 1.2 \pm 0.1 \times 10^{-8}$. As explained in Ref. 123, the source of systematic error in the rate constant was the lack of overlap between $\rho(\lambda^{\perp}|\lambda_0)$ and $P(\lambda_B|\lambda_0;\lambda^{\perp})$. Successful transitions require large momentum when exiting the initial basin, and few to none of the trajectories captured at $\lambda_0$ had the requisite momentum. Even successful transition paths from FFS exited the initial state with lower momenta compared with other methods, resulting in a low estimate of the rate constant. This also resulted in the unphysical result that the momenta of transition paths from FFS were not symmetric about the barrier.

We perform cFFS with the above potential at identical conditions as Ref. 123. The two variables for cFFS are the position ($r$) and momenta ($p$). The basin simulation is performed with 4 million steps. We place interfaces adaptively, collecting ∼2,000 configurations per interface. As in Ref. 123, we initiate 20,000 trajectories from each interface. cFFS resulted in shooting from 8 interfaces, compared with the 7 interfaces used in Ref. 123. The average rate constant from three cFFS trials was $k_{AB}^{\text{cFFS}} = 2.0 \pm 0.1 \times 10^{-7}$, slightly underestimating the EPF rate constant from Ref. 123. The TPE and configurations collected at each interface from cFFS are shown in Fig. 5.4. Paths exit the initial state orbiting the basin and acquiring more kinetic energy until they are able

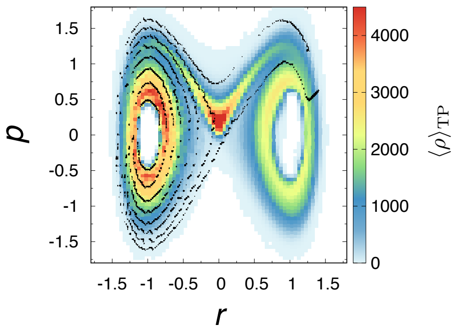

Figure 5.4: cFFS on 1D potential with one position coordinate ($r$) and one momentum coordinate ($p$). Initial basin $A$ is $r < 0$ minima and final basin $B$ is $r > 0$ minima. Configurations collected at each interface are shown as black points. Color map shows the TPE sampling.

to escape. Their momenta then approaches zero as they cross through the transition state, before accelerating towards and orbiting into the final state. Consistent with theoretical expectations, the TPE generated by cFFS is symmetric about the barrier. We also tested cFFS with less sampling. Even with a twenty-fold reduction in sampling (1000 trajectories, 100 configurations per interface), the rate constant calculated with cFFS is $k_{AB}^{\mathrm{cFFS}} = 2.6 \pm 0.7 \times 10^{-7}$ and the TPE remains symmetric about the barrier.

These results demonstrate the potential for using cFFS to study transitions with important momenta variables. Though the above test case represents an extremely simple analytical model, it demonstrates the advantages of cFFS in such scenarios. If an important momenta variable is known for a transition, cFFS allows the basins to be separated with a position coordinate and the momentum coordinate can be used to help drive the transition.

## 5.5 Demonstration on alanine dipeptide

In keeping with tradition, we close by demonstrating cFFS on the $\mathrm{C}_{7\mathrm{ax}}$-to-$\mathrm{C}_{7\mathrm{eq}}$ conformational change in alanine dipeptide in vacuum. Details of the simulations and cFFS are reported in the SI. $\phi$ and $\psi$ backbone dihedral angles were used as CVs for cFFS. The progression of cFFS is shown in Fig. 5.5(a). Starting from the $\mathrm{C}_{7\mathrm{ax}}$ basin centered near $\phi = 60°$ and $\psi = -30°$, cFFS drives the system to the $\mathrm{C}_{7\mathrm{eq}}$ basin defined by $-94° < \phi < -60°$ and $12° < \psi < 90°$. The shape of the interfaces shows that $\phi$ plays the larger role in the transition and reveals the location of the primary transition tube. The transition rate constant predicted by cFFS ($k_{AB}^{\mathrm{cFFS}} = 5.0 \times 10^{6}\ \mathrm{s}^{-1}$) compares favorably with straightforward simulation ($k_{AB}^{\mathrm{SLD}} = 4.8 \times 10^{6}\ \mathrm{s}^{-1}$). In Fig. 5.5(b) we show

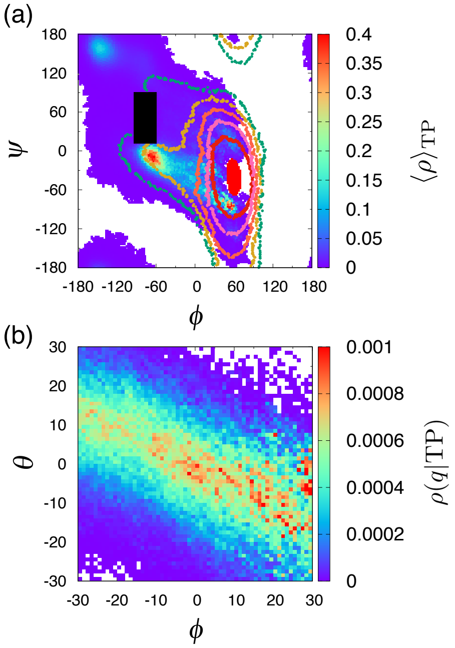

Figure 5.5: cFFS for alanine dipeptide in vacuum. (a) Initial and final states are shown as red and black regions, respectively. Configurations collected at $\lambda_0$, $\lambda_1$, $\lambda_2$, and $\lambda_3$ are reported as red, pink, salmon, gold, and green points, respectively. Color map represents the TPE sampling. $\phi$ and $\psi$ angles are reported in degrees. (b) Correlation between $\phi$ and $\theta$ in the TPE. Color map represents the TPE density of states.

the relationship between $\phi$ and another dihedral angle, $\theta$, in the TPE. It has been shown that $\theta$ is part of the reaction coordinate [260, 247]. cFFS captures the proper relationship between $\phi$ and $\theta$ even though $\theta$ is not one of the CVs used during cFFS [247, 249].

## 5.6 Discussion

cFFS helps overcome a few challenges posed by FFS. cFFS allows one to try multiple CVs simultaneously. This is beneficial for systems where investigators have some a priori insight into the CVs that are expected to play a role in the transition, but a detailed analysis of the mechanism is missing and the best order parameter remains unknown. By using multiple CVs simultaneously and enforcing constant flux along an interface, the method can alleviate issues associated with poor overlap between $\rho(\lambda^{\perp}|\lambda_i)$ and $P(\lambda_B|\lambda_i;\lambda^{\perp})$. Of course, it is possible that there are additional important orthogonal coordinates beyond the chosen CVs. This situation could pose sampling challenges for cFFS. Finally, we demonstrated cFFS with a combination of momenta and position based coordinates. This may extend the practical applicability of FFS to more ballistic systems. FFS depends on stochasticity for trajectory divergence between subsequent interfaces, so it will still not be applicable in the limit of fully deterministic dynamics.

cFFS can in principle be extended to a large number of CVs. However, we surmise the method will not scale well to more than three or four CVs. In high dimensional space, the area through which trajectories can cross an interface will become exceedingly large. From a practical standpoint, this will make it difficult to maintain the constant flux condition. From an efficiency standpoint, most of each interface will drive the system towards regions of phase space which are irrelevant to the transition of interest. Even if successful transitions are generated, they will probably originate from a tiny subset of the phase points collected at $\lambda_0$ and thus be highly correlated. Challenges associated with scaling to large numbers of CVs are hardly limited to cFFS. A variety of advanced sampling methods, including nonequilibrium umbrella sampling [261, 262] and metadynamics [125] have come across similar problems. One solution is to collapse the reaction coordinate to a single dimension using a string-type approach [247, 262, 258]. The string-type approach will prove difficult to implement in FFS without resorting to an iterative scheme requiring multiple FFS runs, because each path ensemble in FFS is generated sequentially and there is no opportunity to relax the string. Moreover, the string-type approach could defeat one of the benefits of cFFS, which is that it enables exploration of transitions with multiple tubes.

Extending cFFS to large numbers of CVs will thus require alternative approaches. Dimensionality reduction techniques such as isomaps [263, 264] or diffusion maps [265, 266] could be employed to reduce a large number of CVs to two or three reduced coordinates which capture the largest spread in the data. In this manner, multiple transition tubes would hopefully be preserved [264, 252] within the reduced coordinates. Furthermore, several groups are actively working to combine machine learning and advanced sampling methods to identify important CVs on the fly [267, 268, 269, 270, 271]. We are exploring if such methods or variations thereof can be incorporated with cFFS. One challenge to incorporating on the fly identification of reduced coordinates with FFS-type methods is again related to the sequential generation of ensembles. Sampling from the initial basin alone is unlikely to reveal reduced coordinates ideal for studying the transition. As FFS progresses, sampling from each interface ensemble will result in reduced coordinates which increasingly describe the transition. However, FFS requires that each ensemble be visited sequentially, and changing the definition of the reduced coordinates after each ensemble may cause substantial difficulty in maintaining this condition.

Studying rare events in simulations is an important and challenging problem that has spawned the development of many methods in the past decades. Here we restrict our comparison to two methods which use multiple CVs to sample, and calculate rate constants for rare transitions with unbiased dynamics in equilibrium or nonequilibrium systems. Vanden-Eijnden and Venturoli developed a method [272] that calculates the transition rate constants and transition paths from the steady state distribution under the boundary conditions that state $A$ is a source and state $B$ is a sink. The space between the stable states is tiled into enclosed Voronoi cells and parallel simulations are performed in each cell. The steady state flux and probability distribution can be estimated from the time spent in each cell and exchange between cells. Like cFFS, the method is applicable to equilibrium as well as nonequilibrium systems and does not require that $A$ and $B$ be well separated in both variables. Since each parallel path is restricted to a single cell, the method may prove advantageous compared with cFFS for systems with metastable intermediates. The method does not provide direct access to dynamical transition paths, although, in principle, transition paths could probably be reconstructed with an extensive bookkeeping scheme. It is not immediately apparent which method would be better for systems with slowly decorrelating transition paths.

As mentioned in the introduction, Borrero and Escobedo [144] developed a method to optimize the FFS order parameter through a series of FFS runs. The approach uses committor information obtained from the prior FFS run to identify the best order parameter from a set of specified CVs. The procedure can be repeated until TPE sampling or the optimal order parameter converges. Like cFFS, the procedure in Ref. 144 allows FFS to be used in situations where there are a number of possible CVs. Since the FFS runs themselves are performed along a single order parameter (which may be a linear or nonlinear combination of multiple CVs), there is no limitation to the number of CVs which can be tested. For certain systems this may represent a substantial advantage over cFFS, which in current form is practically limited to three or four CVs. Unfortunately, the method presented in Ref. 144 requires multiple (often expensive) FFS runs. Additionally, given the sensitivity of FFS sampling to the choice of order parameter in the presence of multiple transition tubes, we suspect cFFS will perform better for such systems.

Lastly, we would like to comment on the possibility of combining a cFFS-type approach with other path sampling methods. At the most basic level, cFFS divides CV space into a fine grid to help define regions of phase space and interfaces between those regions with arbitrary shape. In cFFS, criteria for boundary identification were selected to meet the needs of FFS – a minimum number of total first crossings and constant average flux along the interface to avoid biasing the system to proceed in one direction over another. It is easy to imagine modifying the boundary identification criteria for other applications. Within the family of FFS approaches, it may prove fruitful to combine the approach of Borrero and Escobedo [144] with a cFFS-type approach for interface definitions. This could allow interfaces with any arbitrary shape which could better reproduce the committor function. Transition interface sampling is less sensitive to the definition of order parameter [136, 123]. However, a procedure has been proposed to optimize interface placement given the order parameter [273]. This criterion for optimal interface placement could be combined with a cFFS-type approach for dividing the CV space for transition interface sampling.

## 5.7 Concluding remarks

We described cFFS, a method to sample rare event transitions along multiple CVs simultaneously. cFFS uses automated nonlinear interface placement and reveals on the fly the evolution of CVs during a transition. cFFS was tested with two CVs, but in principle, it can be extended to three or more. In practice, extending cFFS in current form to more than three or four CVs may prove challenging. The stable states only need to be separated in a combination of CVs, which may change nonmonotonically between the stable states. We introduced a criterion of constant flux along each interface to prevent biasing TPE. cFFS results in correct estimates of the transition rate constants and TPE sampling on several 2D PESs and the $C_{7ax}$-to-$C_{7eq}$ transition in alanine dipeptide in vacuum. We additionally demonstrated cFFS on 1D analytical potential using one position coordinate and one momenta coordinate. cFFS substantially improved upon FFS results on the same potential, where only the position coordinate was used an the order parameter. On the 2D PESs, cFFS performed particularly well for systems with hysteresis or multiple transition tubes. cFFS with multiple suboptimal order parameters consistently outperformed FFS with a single suboptimal order parameter. Since optimal order parameters are not known in most applications of FFS, cFFS with two or more suboptimal order parameters will be beneficial for studies of complex systems such as macromolecular conformational transitions and crystal nucleation.

## Chapter 6

# A generalized deep learning approach for local structure identification in molecular simulations[1]

### 6.1 Introduction

Molecular simulations have become an indispensable tool in investigations of wide-ranging phenomena in physics, chemistry, biology, and engineering. One of the primary goals of molecular simulations is to relate microscopic behavior to macroscopic observable properties of a system. As such, local structure (i.e., spatially local arrangements) of atoms and molecules is often of interest. In principle, the raw output from molecular simulations—the position coordinates of each atom in the system for the duration of the simulation—includes all the necessary information to quantify local structure. However, this immense quantity of data inevitably requires further analysis to extract meaningful insights into system behavior. Quantitative measures of local structure are thus imperative in analysis of molecular simulations.

[1] Material for this chapter adapted from Ref. [274]

The current approach is to calculate some mathematical quantity that is a function of the atomic positions. These functions are referred to as ‘order parameters’ since they often track structural order present in a system. In some cases, order parameters can be trivial to implement and intuitive to understand (e.g., distance between an ion pair [243]). However, many types of local structure have more subtlety (e.g., solvation environments, crystal polymorphs) and require substantially more complex order parameters. Though there are some general approaches that have proven successful for certain types of systems [275, 276, 277, 278], order parameter development is a highly challenging and non-trivial endeavor. This difficulty is evidenced by the fact that demonstrating new order parameter(s) is often itself worthy of a full paper [279, 280, 194, 191, 281]. Given the difficulty of order parameter development, it is unfortunate that most order parameters only distinguish a small set (i.e., <3) of physical structures. Development and/or implementation of multiple order parameters is often required if it is necessary to distinguish between additional structures. These challenges hinder progress in applying molecular simulations to study novel structures and systems.

The widespread success of machine learning has prompted researchers to apply techniques from this field in capacities that span nearly every aspect of molecular simulations. Several groups are actively working to directly combine machine learning with advanced sampling methods [269, 282, 283]. Others have used machine learning to develop force fields [284, 285], for coarse-graining [286, 287], to identify reaction coordinates [210], and to extract trends in results by clustering similar structures [288, 289, 290]. There are a few previous efforts to use machine learning for structure identification in molecular simulations [291, 288, 292, 293, 294, 295]. In general, such approaches seem promising given the widespread success of machine learning in tasks such as computer vision (e.g., image recognition, segmentation, etc.) [296, 297, 298]. Structure identification in simulations is in many ways similar to computer vision tasks, where simple units of patterns or structures (e.g., distances and angles between atoms) combine in specific larger patterns to form some structure (e.g., crystal type, macromolecular secondary structure).

The primary challenge in applying machine learning to structure identification in molecular simulations is determining the appropriate input features. Several interesting approaches have relied on preprocessing the output from molecular simulations. Geiger and Dellago [291] trained a neural network to identify crystal structures in Lennard-Jones (LJ) and water systems by processing the raw output from molecular simulations through system-specific symmetry functions [284].

Other approaches for structure identification in molecular simulation have first processed the simulation output with spherical harmonics [293, 295] or other carefully engineered features [294, 295]. Panagiotopoulos and co-workers took a different approach, using neighborhood graphs and diffusion maps [265, 266] to discover, cluster, and classify crystal structures and identify relationships between crystal types without explicit training of each structure [288, 292]. In general, the previous attempts to apply machine learning for structure identification in molecular simulation require extensive preprocessing [291, 294] or complex and computationally expensive methods [288, 292]. In particular, preprocessing data can require extensive system-specific parameterization and risks limiting the methods to specific types of local structure.

Our goal is to develop a simple and straightforward machine learning approach to distinguish between any number of distinct local structures that appear in molecular simulations. These local structures could be representative of a local crystalline environment, biomolecular conformation, relative arrangement of ligand and binding site, or, in principle, any other structure. The key feature is that the structures must be distinct. In the context of machine learning, this represents a classification problem. Neural networks, a cornerstone of deep learning, have been increasingly used for classification and segmentation tasks in a plethora of diverse applications. Deep learning offers a suitable approach for classifying structure in a molecular simulation because of its ability to extract high level representations from raw features of large datasets [299]. However, there are challenges to directly applying neural networks to molecular systems. Molecular systems contain physical symmetries that should be preserved: symmetry with respect to global translation, global rotation, and exchange of chemically identical atoms. In other words, the classification should remain identical upon applying any of those operations to a structure. One approach to handling these symmetries is through preprocessing (e.g., symmetry functions) [284]. However, this can require system-specific tuning and/or *a priori* knowledge [279]. We aim to create a procedure that requires minimal preprocessing or tuning by training a neural network on data that is as close as possible to the raw output from molecular simulations. The approach should be easy to apply, able to distinguish between multiple different structures, and applicable to a range of systems studied in molecular simulations.

Using ideas from computer vision [300], we implement a network that is designed to operate directly on sets of points—i.e., the output of molecular simulations. We apply our approach to differentiate between the liquid phase and various crystal structures. This choice is motivated by several factors. (1) Molecular simulations are widely used to study phase equilibrium and phase transitions in a myriad of substances. Crystal structure identification is required in studies of crystal nucleation [17, 131], crystal growth/dissociation [301, 302], crystal defects, and crystal grain boundaries [303]. (2) It is important to have a local (rather than global) measure of the crystal type for all of the previously listed problems, (3) there are many types of crystal structures, providing a rich test bed for our approach, (4) more recent studies have focused on heterogeneous nucleation (i.e., in the presence of an external interface), where existing order parameters are often unable to identify the crystal type of atoms nearest to the external interface [214], and (5) successfully demonstrating the method for crystal structure identification in multiple types of systems suggests it should be broadly applicable to any assembly process (e.g., crystallization, protein folding, etc.) that involves the formation of multiple distinct structure types that must be identified.

The methodology is described in Sec. 6.2. Results from three different types of systems (LJ, water, and mesophase) are presented in Sec. 6.3. The method is extended to identify hydrophobicity on surfaces in Sec. 6.4. Concluding remarks and future directions are provided in Sec. 6.5. For brevity, all simulation details are provided in Appendix C.

## 6.2 Methods

In machine learning, point clouds are a data structure that consist of a set of points in 3D space. Each point is represented as $(x, y, z)$ coordinates, and a single point cloud consists of a collection of individual points. Point clouds are notoriously challenging to work with because of their irregular data structure. One common way to handle point clouds is to convert the data structure into a regular 3D voxel or a collection of images so it can be processed using existing methodologies that handle regular, lattice-like data, such as 2D or 3D convolutional neural networks [304, 305]. However, when dealing with a large number of data points, these transformations create needlessly voluminous data. Computation times for model training and evaluation can quickly exceed practical limits and these data preprocessing transformations can also obscure natural invariances within the data [300]. We elect to use a recently developed deep learning model known as PointNet, which directly processes point clouds without preprocessing [300]. The input to PointNet is a single point cloud (i.e., set of points) while the output is a class label for classification or a per point label for segmentation.

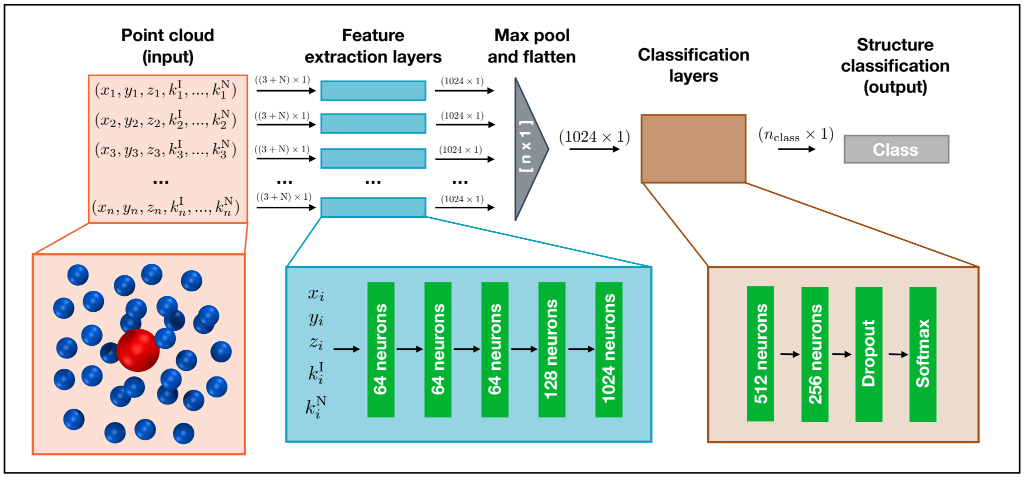

Figure 6.1: **PointNet Architecture** The PointNet takes $n$ points of $3 + N$ dimensions (i.e., a single point cloud) and passes them through feature extraction layers, a max pooling layer, and finally classification layers. The feature extraction is comprised of five dense layers that share weights across each point. The max pooling layer applies a symmetric function with a $n \times 1$ filter, reducing the features to just $1024 \times 1$ dimensions. Two dense layers and a softmax layer are used to determine the final classification of the point cloud.

Our goal is to classify local structure in simulations. Since the raw output from molecular simulations is a point cloud, we use a PointNet to classify structures found in point clouds composed of spatially local atoms. The classification can describe a collective property of all the atoms in the point cloud (e.g., the conformation of a small molecule could be classified from the coordinates of its atoms) or the classification can be projected back onto the central atom to provide a descriptor of the local environment that the central atom is embedded within (e.g., local crystal structure).

### 6.2.1 PointNet structure

We choose to use the basic setting of the PointNet architecture, which consists of a section of shared dense feature extraction layers, a max pooling operation, followed by a section of densely connected layers. A schematic of the network structure is shown in Fig. 6.1. The PointNet takes a point cloud comprised of $n$ points as input. Each point $i$ is composed of $x_i$, $y_i$, and $z_i$, to which 0–N additional features can be appended $(k_i^1, \ldots, k_i^{\mathrm{N}})$. Before entering the feature extraction layers, the point cloud is randomly rotated around the $x$, $y$, and $z$ axis. This transformation is required to train the network to be invariant to global rotation. Each point is *identically* and *independently* passed through the feature extraction layers. The feature extraction is a multilayer perceptron

(MLP) network composed of five dense layers with 64, 64, 64, 128, and 1024 neurons, respectively. It is important to note that the weights of the feature extraction networks are shared across each point, similar to a convolution operation. Following the feature extraction, each point has been transformed from being described by 3 + N features to 1024 features. Furthermore, each point has undergone the same mathematical transformations, regardless of the input ordering of the points.

It is necessary to combine features from the different points in order to reach an overall classification, but a strategy must be applied to make the model permutation invariant of the input order. The PointNet implementation we use employs a symmetric function that aggregates information from each point, irrespective of initial point order. The idea is to approximate a general function (i.e., classification) by combining feature transformations and a symmetry function as follows:

$$f(\{p_1, ..., p_n\}) \approx g(h(p_1), ..., h(p_n)) \tag{6.1}$$

where $p_1, \ldots, p_n$ are the points in the point cloud, $h$ is the feature extraction MLP and $g$, the symmetry function, is the max pooling function. The max pooling function uses a filter of size $[n, 1]$, where $n$ is the number of points. Thus, the max pooling function selects, *for each feature*, the value that is most highly activated, regardless of the point that it comes from. The output from this operation is symmetric with respect to changing the order of the input points. The combination of the shared feature extraction MLPs and this max pooling layer allows us to input a set of spatial coordinates describing a molecular structure, and maintain the invariance to exchange of atoms. The output of the max pooling layer is fed to two fully connected layers with 512 and 256 neurons, respectively, which is fed to a dropout layer (keep probability 0.7) and finally a softmax layer for classification.

All fully connected layers, including the layers in the feature transformation, use batch normalization [306] and the rectified linear unit (ReLU) nonlinear activation function. We employ the cross entropy loss function and use the Adam Optimizer [307] with learning rate 0.001 and default parameters. The network was implemented using the Tensorflow [308] Python API.

Results should be invariant to global translations and global rotations of the point cloud, as well as the exchange of atoms (i.e., the order of points in the point cloud). Invariance to global translation is handled by shifting the point cloud such that some selected central atom is always located at $(0, 0, 0)$. The invariance to exchange of atoms is handled by the structure of the PointNet.

Invariance to global rotations is handled by the random rotation that is applied to each point cloud before it enters the network.

### 6.2.2 Local crystal structure identification

One common approach in crystal structure identification is to determine the crystal structure of each atom from its local environment. We apply the PointNet in a similar manner to determine the crystal environment of every atom in the system. Each point cloud is created by the positions of atoms within some cutoff distance, $r_{\text{cut}}$, of a central atom, $\mathsf{a}$, and used to classify the crystalline environment of $\mathsf{a}$. Standard periodic boundary conditions are applied when calculating the neighbors of $\mathsf{a}$. Each point cloud is translated such that the central atom, $\mathsf{a}$, is located at $(0, 0, 0)$. This step preserves the symmetry of the network output with respect to global translations of the point cloud. The coordinates of the atoms in the point cloud are scaled such that the closest atom is always at distance of 1.0. The PointNet requires a fixed number of points, $n$, in the input point clouds. However, the number of atoms within a given cutoff distance of a central atom varies from sample to sample. To handle this problem we pre-selected a set point cloud size for a given cutoff distance. If a sample contained more than $n$ points, the furthest points from the central atom were removed from the point cloud until only $n$ points remained. If the sample contained fewer than $n$ points, the point cloud was padded with $(0, 0, 0)$ points. The number of points in the point cloud, $n$ was selected as two standard deviations above the mean number of points within the cutoff distance.

### 6.2.3 Training and testing methodology

The PointNet is trained on samples generated from simulations of pure crystal phases. This approach provides easy ground truth labels for the samples. $10^4$–$10^5$ samples were collected from simulations of each pure phase. We ensured that there were an equal number of samples from each phase. Samples were randomly divided into a training and testing portion, with 80% of samples in the training set and the remaining 20% of samples belonging in the test set. Each time a point cloud is processed through the network, it is given a different random rotation around its $x$, $y$, $z$ axes. Thus, the network is forced to learn the invariance to global rotations of the point clouds. The network was trained for 100 epochs with each batch containing 64 point clouds.

The test set is sampled from the original data distribution. To more rigorously check for model generalization, we devise a 'hold-out' set in addition to the test set for some systems. The hold-out set contains the same crystal phases as the training and test data, but the samples are taken from simulations performed at temperature and pressure conditions that are interpolated between the conditions used for training and testing. The hold-out set is designed to ensure that the network has indeed learned the emergent features of the crystal structure and generalizes to conditions not explicitly seen by the network during training.

## 6.3 Results and Discussion

### 6.3.1 Lennard-Jonesium

We first tested the ability of the PointNet to classify phases formed from Lennard-Jones (LJ) particles. The network was exposed to four classes of structures during training, namely, liquid, fcc, hcp, and bcc. To provide a robust test of the PointNet, the trained network was tested on point clouds that were taken from $(T, P)$ conditions that the network was never exposed to during the training phase. The $(T, P)$ conditions were selected such that they interpolated between the $(T, P)$ conditions of the training data. Details of training conditions are specified in Appendix C. Each point in the point clouds was comprised only of $(x_i, y_i, z_i)$—no additional features were appended (i.e., $\mathrm{N} = 0$ in Fig. 6.1).

Results are reported in Fig. 6.2. Fig. 6.2 shows both the classwise accuracy of the network and which class the network tends to (incorrectly) select when it misclassifies a sample. The abscissa of each panel in Fig. 6.2 lists the true class of each sample. The bar reports the classification selected by the PointNet. Note the use of broken ordinate axes to highlight both low and high percentage regions. The cutoff radius ($r_{\text{cut}}$) for the point cloud was increased from $r_{\text{cut}} = 1.5$ for Fig. 6.2(a) to $r_{\text{cut}} = 2.0$ for Fig. 6.2(b) and $r_{\text{cut}} = 2.6$ for Fig. 6.2(c). With increasing $r_{\text{cut}}$, the number of points in the point cloud increases. The point clouds for $r_{\text{cut}} = 1.5$, $r_{\text{cut}} = 2.0$, and $r_{\text{cut}} = 2.6$ were 16, 43, and 83 points, respectively. Not surprisingly, the accuracy of the PointNet increases with increasing cutoff radius, from 95.2% for $r_{\text{cut}} = 1.5$ to 99.2% for $r_{\text{cut}} = 2.6$. A previous effort [291], which used machine learning to classify LJ phases, performed slightly better than 95% accuracy with a cutoff of 2.6. The method employed a substantially smaller network (∼3,000 trainable parameters vs. ∼800,000 in our network), but preprocessed the raw input data through user-defined and parameterized symmetry functions. Larger cutoff radii increase the number of points in the point clouds and enable PointNet

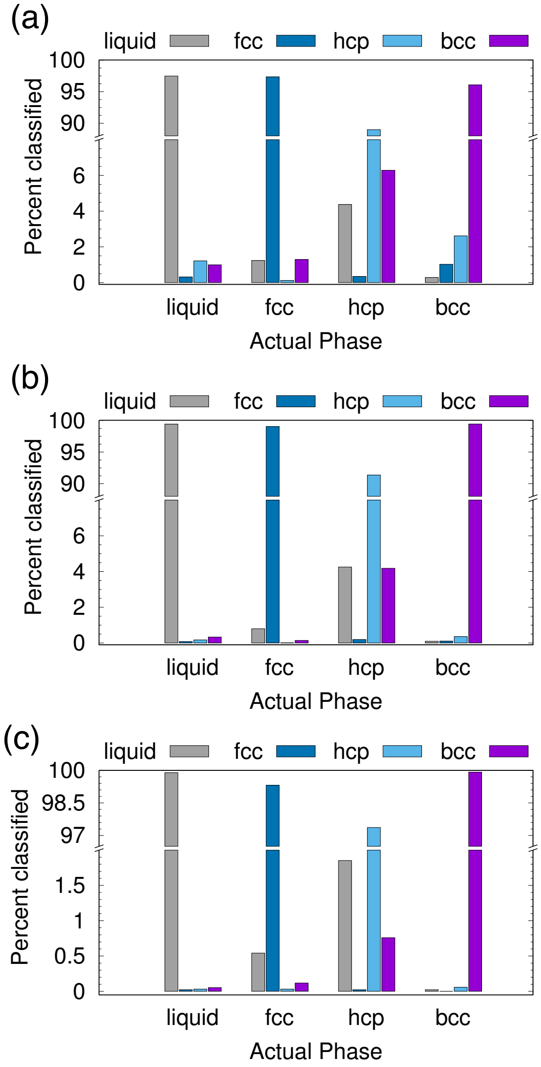

Figure 6.2: Classification choices of a PointNet trained on four phases of a LJ system with (a) $r_{\text{cut}} = 1.5$, (b) $r_{\text{cut}} = 2.0$ and (c) $r_{\text{cut}} = 2.6$ distance units. Note the ordinate axis of panel (c) differs from (a) and (b). Overall accuracy is 95.2%, 97.7%, and 99.2% for $r_{\text{cut}} = 1.5$, $r_{\text{cut}} = 2.0$, $r_{\text{cut}} = 2.6$, respectively.

to identify longer range repeating patterns in the crystal structures. These patterns presumably become more distinct and easier to differentiate with larger point clouds. Furthermore, increasing the cutoff reduces the probability that a sample from one phase will spontaneously adopt, through thermal fluctuations, a configuration that appears identical to a different phase.

#### 6.3.1.1 Structure identification in crystal seeds

To further test our method, we focused on the crystallization aspect. We generated LJ systems comprised of crystalline nuclei surrounded by a liquid bath. Simulations were performed for systems with three different sizes of the initial crystalline seed. The dynamics were propagated with molecular dynamics (MD) at a temperature below the melting point. If the initial nucleus size is above the critical size, the crystalline seed grows to encompass the entire system; if it is below the critical size, the crystalline seed melts. These types of simulations are used as part of the seeding method [63, 64], which is used to predict crystal nucleation rates. The definition of the exact size of the crystal seed is one of the largest sources of uncertainty in the method [64]. The test case enabled us to explore several questions regarding the behavior of the PointNet. How would the network perform outside of pure phases? Would the network provide reasonable classifications for atoms near the boundary of solid and liquid phases? Would the network be able to identify defects within the solid phase? How would classifications be affected by changing the cutoff radius?

We tested PointNet trained with the three different cutoff radii ($r_{\text{cut}} = 1.5$, $r_{\text{cut}} = 2.0$, $r_{\text{cut}} = 2.6$). The identity of each atom in the system was calculated every 0.1 time units. The crystalline nucleus was identified at every step as the largest cluster of connected solid atoms (all fcc, hcp, and bcc atoms are considered solid) in the system. Two atoms are connected if their distance is within the distance to the first minimum in the liquid radial distribution function.

The evolution of seed sizes is reported for the three different seeds (initial sizes 140, 211, 372 in Fig. 6.3(a). Following an equilibration where the seed is held rigid, the calculated seed size is reported with time for each value of $r_{\text{cut}}$. Larger cutoff radii result in smaller predicted seed sizes. This result is not particularly surprising; from Fig. 6.2, it would be expected that more liquid atoms surrounding the seed will be incorrectly classified as solid for the smaller values of $r_{\text{cut}}$. These atoms, incorrectly classified as solid, are added to the surface of the largest solid cluster, resulting in a larger overall cluster size. This effect is in many respects similar to the results found from using a stricter classifier with traditional order parameters [309]. Effectively the PointNet acts as a stricter

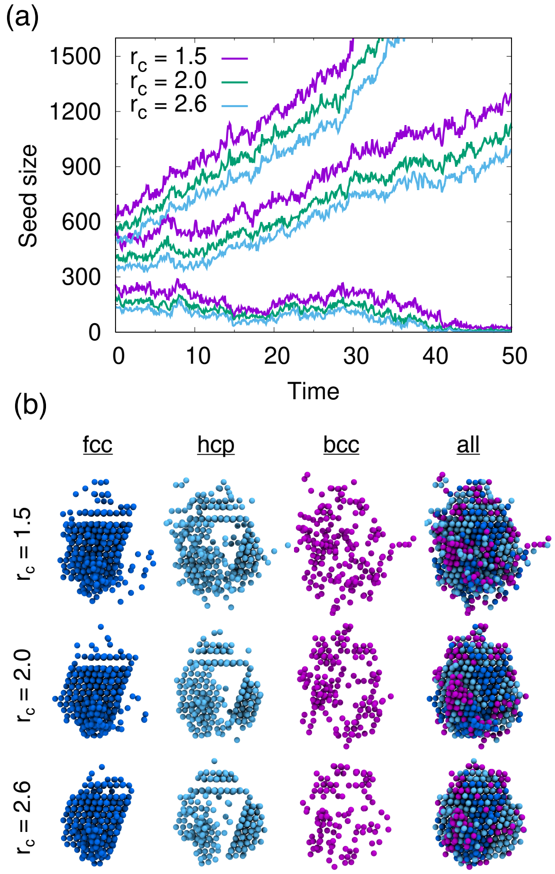

Figure 6.3: Growth and dissociation of crystalline seeds identified by a PointNet in LJ systems. The behavior of the overall seed size with time is shown in panel (a). Snapshots of a seed at one time point are shown in panel (b) to show the variation in classification with changing cutoff distance.

classifier when there is a larger cutoff radius.

One important question to consider is how the classifications of the solid atoms in the cluster change with increasing cutoff radius. Ideally, the overall structure and composition of the crystalline nucleus identified by the PointNet would be relatively insensitive to the choice of $r_{\mathrm{cut}}$. Snapshots for one crystalline nucleus at a single time are reported in Fig. 6.3(b). The first three columns show the fcc, hcp, and bcc atoms that belong to the largest cluster of solid atoms. The fourth column overlays all the classes to show the complete crystalline nucleus. From the rightmost column it is apparent that the overall nucleus size decreases with increasing cutoff. However, the atoms that are removed from the crystalline cluster are surface atoms that do not appear to display substantial crystallinity. Encouragingly, the first two columns show that classifications of atoms in the core of the cluster is insensitive to changing the cutoff radius of the PointNet. In particular, the network consistently identifies a single layer of hcp atoms stacked between layers of fcc atoms as well as clear hcp layers growing along several edges of the nucleus. It is not particularly surprising to find that the size of the crystalline cluster decreases as the PointNet becomes a stricter classifier (i.e., with increasing cutoff). However, it is encouraging to find that the identities of atoms with the core of the crystal remain relatively consistent with different cutoff values.

### 6.3.2 Water systems

Ice and hydrate nucleation are particularly active areas of research [53, 54, 132, 131]. One of the largest challenges in this field is structure identification. Thus, we next tested the ability of the PointNet to classify the phases of water molecules. The PointNet was trained on eight different phases, including liquid, five ice phases, and two hydrate phases. Unlike the LJ systems, water is comprised of two different atom types. Historically, most methods for classifying ice structures rely only on the positions of the oxygen atoms. However, there may be additional information to be gained by including the positions of the hydrogen atoms. We explore both options.

#### 6.3.2.1 Classification of water phases using only oxygen atoms

At first, only the positions of the oxygen atoms were used in the point clouds. In this case, all points in the point cloud are identical (with respect to atomic identity), therefore the points are completely described by $(x_i, y_i, z_i)$ and no additional features are appended. Results are shown in Fig. 6.4 using a cutoff radius of 0.6 nm. The point clouds contain 43 points. The PointNet does exceedingly well at distinguishing the liquid, ice phases, and hydrate phases. Once again, the liquid is sometimes (<1.5%) identified as one of the solid phases. Our results for the LJ systems suggest that misclassification could be reduced by further increasing the cutoff radius. In particular, it is worth noting that the PointNet performs better on water systems compared with LJ systems with the same number of points in the point cloud (for LJ systems, $\mathrm{r_{cut}} = 2.0$ has 43 points). We conjecture that this observation is related to the open network structure of water [310]; a water molecule only has an average of four first neighbors whereas LJ systems have closer to 12. Thus, for the same 43 points the PointNet can evaluate further neighbor shells for water relative to the LJ systems.

The network has the greatest difficulty distinguishing between the two hydrate phases. Though the network is able to clearly distinguish these phases from the liquid and ice phases, ~2% of sII samples are classified as sI and ~3% of sI samples are classified as sII. The hydrate phases have, on average, fewer points within the cutoff distance; it seems quite plausible that this difference is the source of greater error in classifying the hydrate phases. Despite minor difficulties with the hydrate phases, the overall accuracy is still superior to previous attempts to classify multiple ice structures with neural networks [291]. An approach that was published during the writing of this paper achieved higher classification accuracy but only distinguished between the ice Ih and liquid phases [295].

#### 6.3.2.2 Classification of water phases using oxygen and hydrogen atoms

Next, we investigate the effect of including positions of the hydrogen atoms in the point cloud. Each water molecule is still classified as a certain phase. It does not seem sensible to classify the hydrogen of a water molecule as belonging to ice-Ih and the oxygen of the same water molecule as belonging to ice-Ic. Therefore, we generate a point cloud centered around each oxygen atom. However, instead of the previous approach where the points in the point cloud were only comprised of the oxygen atoms, we now also include the positions of the hydrogen atoms. Given the distinct hydrogen-bonding patterns in water, we hypothesize that including the hydrogen atom positions in the point cloud will improve classification. The network structure presented in Fig. 6.1 indicates that additional features $(k_i^{\mathrm{I}}, \ldots, k_i^{\mathrm{N}})$ can be appended to $(x_i, y_i, z_i)$ for point $i$. We use this ability to include the atomic identity of each point. Two features, $k_i^{I}$ and $k_i^{II}$ are added to each point. The atomic identity is one-hot encoded. Oxygen atoms are encoded as $(x_i, y_i, z_i,\ 1,\ 0)$ and hydrogen

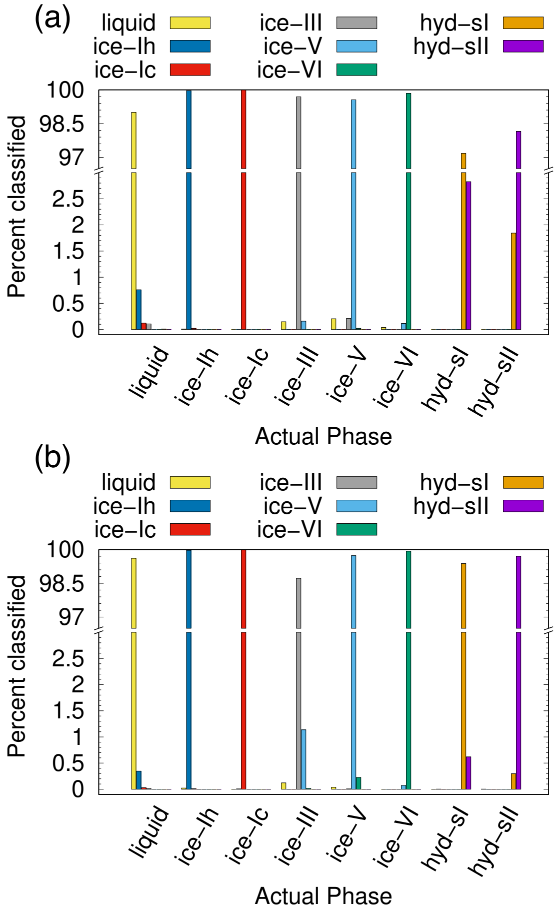

Figure 6.4: Classification choices of a PointNet trained on eight water phases with a cutoff of 0.6 nm. (a) Using only oxygen atom positions in the point cloud (overall accuracy is 99.1%), and (b) using both oxygen and hydrogen atom positions in the point cloud (overall accuracy 99.6%).

atoms are encoded as $(x_i, y_i, z_i, 0, 1)$. This distinction should enable the network to recognize the difference between oxygen and hydrogen atoms during feature extraction (see Fig. 6.1).

The addition of the hydrogen atoms with atomic identity improved the overall accuracy from 99.1% to 99.6%. Though this seems like a relatively minor improvement in the accuracy statistic, it dramatically improved the classification of the sI and sII hydrate phases (Fig. 6.4(b)) and in fact reduced the total misclassification rate by over 50%. In certain cases it can be extremely important to minimize particle misclassifications. In particular, it can be important to minimize liquid particles that are misclassified as solid [63, 64]. The results for the PointNet when hydrogen atoms are included have nearly as low misclassifications (0.4% vs. 0.25%)[63] as the strictest order parameter for identifying ice structures [279]. This result is despite the fact that the strictest order parameter was used to distinguish between only two phases while the PointNet is distinguishing between 8 phases. Furthermore, the accuracy of the PointNet could no doubt be increased further by increasing the cutoff radius.

#### 6.3.2.3 Nucleation at solid interfaces

Significant effort has recently focused on heterogeneous ice nucleation [132, 132, 55, 60, 157, 181], that is, ice formation that occurs in the presence of an external interface. We thus decided to test the ability of our PointNet to classify the types of ice that form on surfaces. In initial tests (data not reported), the PointNet was unable to correctly identify the identities of the water molecules in the layer of water nearest to the interface, even though the second layer of water and above were correctly classified. The reason the PointNet was unable to correctly identify the identity of interfacial water molecules is that the point clouds for these samples were effectively cut in half, from a sphere to a dome shape. No water molecules penetrate into the surface, and thus, the point clouds for water molecules nearest to the surface only have points from the water molecules in the direction opposite from the surface.

Classification of interfacial molecules with the PointNet thus presents a unique challenge. There are a few potential approaches to solve this problem. One could generate interfacial training data for each phase. These data could be generated by simulating each phase in contact with a solid surface or at an interface with a vacuum. Unfortunately, both of these options present a range of further complications. In the first case, since the structure of water near the surface may be affected by the chemical composition and structure of the surface, it would be necessary to simulate near several different surfaces to achieve any substantial model generalizability. In case of a vacuum interface, the layer nearest the vacuum would likely melt [311], or at least deform, thus requiring position restraints (or some similar approach) to maintain the crystal structure. In both cases, there is the question of which crystal plane to simulate as the exposed surface. In most cases where researchers are actively studying heterogeneous crystal nucleation, the crystal plane that nucleates on a surface is unknown *a priori*. In all likelihood, it would be necessary to simulate multiple crystal faces for each phase. Any of these complications add substantial complexity to the structure identification process because they require significant additional simulations.

The ideal scenario is to train the PointNet to correctly identify interfacial molecules without requiring additional training data or making any assumptions about which crystal faces are most likely to form at some surface. To this end, we developed and tested the following approach. For each bulk training sample (i.e., a single point cloud with a label) we (1) randomly rotate the point cloud, (2) remove all points with $z < 0.0$ by replacing $(x_i, y_i, z_i)$ with $(0.0, 0.0, 0.0)$, and (3) randomly rotate the point cloud. The label for the sample, $l$, is changed from $l$-bulk to $l$-interfacial. The first rotation removes any memory of the crystal orientation in the simulation box from the simulation of the bulk crystal. Replacing points below the $z = 0$ plane with $(0.0, 0.0.0.0)$ effectively removes those points from the point cloud. This replacement causes no issues during training as we already pad the point clouds with $(0.0, 0.0, 0.0)$ points if there are insufficient atoms within $r_{\text{cut}}$. The final rotation removes the memory of removing all atoms below the $z = 0$ plane. This procedure thus uses the original bulk training data to generate point clouds with a dome geometry. No assumptions are made about the exposed crystal plane or the orientation of the external surface in the simulation box, and all crystal planes are sampled in the procedure without any additional simulations.

The accuracy of the PointNet trained on both bulk and interfacial water phases is reported in Fig. 6.5(a). Only oxygen atoms are included in the point clouds. The first eight phases are the bulk phases and the next eight are the respective interfacial counterparts. The overall accuracy decreased from 99.1% to 92.5%. The decrease in overall classification accuracy is not surprising given that half of the samples (i.e., the interfacial samples) have 50% fewer points in each point cloud. Despite the decrease in overall accuracy, there are several positive features worth noting. Firstly, bulk classification remains extremely accurate. For example, bulk ice phases all remain above 99.5% correctly classified. Secondly, there is no mixing between the interfacial phases and the bulk phases; i.e., an interfacial atom is never classified as bulk and vice-versa. In essence, this means that the

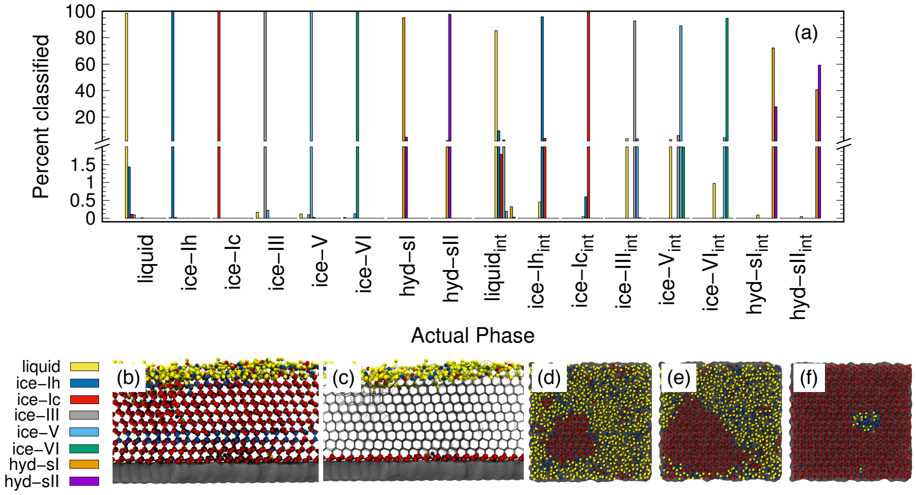

Figure 6.5: (a) Classification choices and (b)–(f) snapshots of classified atoms for a PointNet trained on sixteen water classes (eight bulk and eight interfacial). Panel (b) shows the network identifying a single layer of ice-Ih within ice-Ic. Panel (c) shows atoms that belong to an interfacial class as spheres. Panels (d)–(f) show a top view of the classifications of the first two layers of atoms for a growing ice seed. The color scheme in (b)–(f) corresponds to (a). The external surface is shown in gray. Water oxygens within 0.35 nm of each other are connected by light gray bonds.

PointNet simply struggles to correctly classify the identity of interfacial atoms. It still performs acceptably for interfacial liquid and ice phases. Only the performance of interfacial hydrate phases is particularly poor. One final observation is that the PointNet was trained and tested on all possible crystal planes at the exposed surface. As described previously, this is beneficial in that it makes the extension of the PointNet approach to interfacial systems trivial. There currently exist few methods [288] to identify local crystal structures at interfaces.

Panels (b)–(f) of Fig. 6.5 show snapshots from simulations of the formation of ice at an external surface (gray). Panel (b) shows the classification of each atom in the system. The ice that forms is primarily ice Ic (red spheres) with a single stacking-disordered layer of ice Ih (blue spheres). Consistent with expectations [312], a layer of liquid (yellow spheres) exists at the ice–vacuum interface. Panel (c) only shows atoms classified as interfacial types as spheres. The snapshot confirms the results from panel (a)—no bulk atoms are misclassified as interfacial. Panels (d)–(f) show the growth of an ice nucleus on the surface from a top-down perspective. To highlight the interfacial classifications, only water molecules belonging to the first two layers on the surface are shown. Despite lower accuracy, the interfacial classification appears to perform quite well. The ice nucleus, composed of ice Ic, clearly develops. Very few atoms within the center of the ice seed are classified as any other type. There does appear to be a somewhat greater number of misclassifications in the interfacial liquid, although it is difficult to say with certainty that none of those water molecules are in a locally ice-like environment. We note that the liquid/ice-Ih region in panel (f) is indeed correctly classified. The ice in that region contained a grain boundary between the periodic images of the growing ice crystal that was not fully resolved by the end of the simulation.

### 6.3.3 Mesophases

Mesophase systems have attracted interest in recent years [313, 314, 134, 315, 281, 316] because they can be used to study the behavior of self-assembling materials and block co-polymers. The key feature of such systems is that some molecules remain locally amorphous, even following the transition from a disordered state to an ordered state. The systems are generally composed of two different types of molecules (for simplicity, just consider two particle types, A and B). A rich variety of structures can be formed by tuning the relative size and strength of A–A, B–B, and A–B interactions.

There have been very recent efforts [281] to develop order parameters that distinguish between the different mesophases in order to better understand the behavior (e.g. nucleation) of such systems. Developing order parameters for mesophase systems is particularly challenging because there are a large number of possible phases that can form and, by definition, part of the system is non-crystalline. Previous efforts have resorted to developing different order parameters to distinguish each phase [281]. Here we test the extensibility of the PointNet approach by using these mesophase systems as test cases.

Six phases were simulated (see Appendix C for details): liquid (liq), lamellar (lam), lxs, hexagonal (hex), gyroid (gyr), and body-centered cubic (bcc). Snapshots of the lam, lxs, hex, and gyr phase are provided in Fig. 6.6. Three different approaches to classification are attempted. In all cases, we only attempt to classify the structure of the minor component. Even in non-liquid phases, the major component tends to belong to largely disordered amorphous regions. In the first attempt, both A atom types and B atom types are used as points in the point cloud. However, A types and B types are not distinguished (i.e., no additional features are appended to $(x_i,y_i,z_i)$). We test cutoff values of 2.0 and 3.0. The results for $r_{\mathrm{cut}} = 2.0$ are reported in Fig. 6.7(a). The overall accuracy is $\sim$96%. Increasing the cutoff to 3.0 increases the accuracy slightly, to 97.6%. As can be seen from Fig. 6.7(a), the PointNet is particularly challenged to distinguish the hexagonal and gyroid phases.

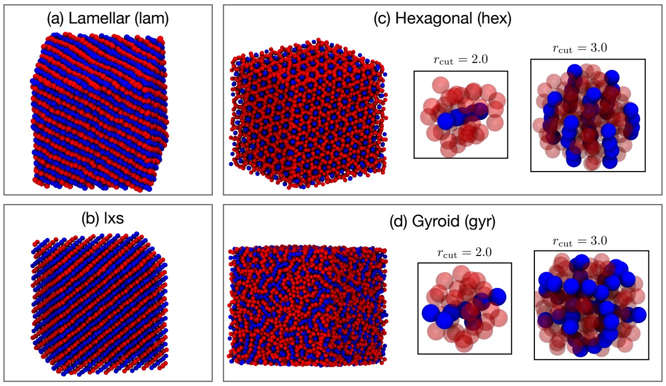

Figure 6.6: Snapshots of four mesophase systems (a) lamellar, (b) lxs, (c) hexagonal, and (d) gyroid. Minor component atoms are shown in blue and major component in red. Example point clouds with a cutoffs of 2.0 and 3.0 are shown for the hexagonal and gyroid phases.

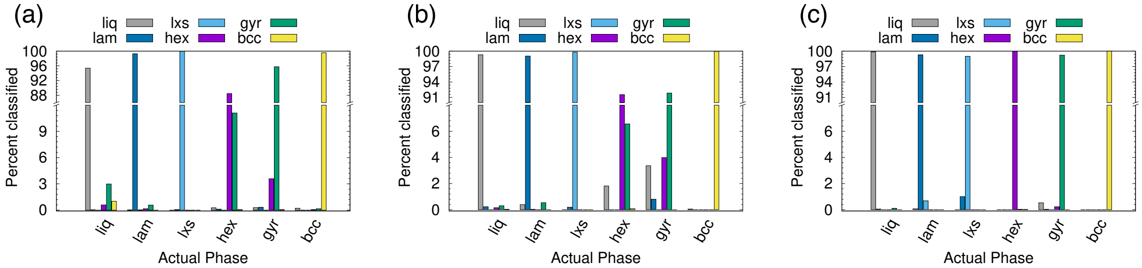

Figure 6.7: Classification choices of a PointNet trained on six phases of a mesophase forming system with a cutoff distance of 2.0. The point cloud included (a) all points, (b) all points with identifying labels, and (c) only points belonging to the crystalline component. Note the ordinate axis of (b) and (c) differ from (a).

This difficulty is explained by the snapshots in Fig. 6.6(c) and (d). With a cutoff of 2.0, the point clouds of the hex and gyr phases appear very similar. The point clouds would look particularly similar in this case, where the type A and B particles are not distinguished.

Next, the atomic identity was added to each point. Similar to the water systems, the atom types were one-hot encoded using two additional features. Results for $r_{\mathrm{cut}} = 2.0$ are reported in Fig. 6.7(b). The addition of atomic identity improved the overall classification accuracy to 97.7%. However, the network still has difficulty distinguishing the hexagonal and gyroid phases.

Given the amorphous nature of the major component in most of phases, it seemed reasonable that the inclusion of the major component might represent unnecessary and confusing information, and that performing classification based only on the positions of the minor component might be a better approach. This method yielded by far the best results (see Fig. 6.7(c)). The overall accuracy was 99.6% with $r_{\text{cut}} = 2.0$. The issues distinguishing between the hexagonal and gyroid phases are largely resolved. If $\mathrm{r}_{\text{cut}}$ is increased to 3.0, the accuracy improves to >99.9%. Using only the minor component to classify the structure of the mesophase system mirrors the approach taken with recent conventional order parameters [281].

## 6.4 Beyond crystal structure identification

To showcase the utility of our method beyond applications in crystal structure identification, we use the PointNet to quantify the hydrophobicity of extended surfaces and proteins. Previous work characterizing surface hydrophobicity used local water density fluctuations or solute affinity over different portions of surfaces to create a spatially resolved measure of hydrophobicity [317, 318, 319, 320]. Recent work found that water orientations near an interface may also be able to predict local surface hydrophobicity [4]. In principle, the PointNet should be able to learn the differences in interfacial water structures near hydrophobic and hydrophilic interfaces. The network is trained to predict if an individual water molecule is in a hydrophobic or hydrophilic environment based upon the point cloud created by neighboring water molecules. Training examples are generated from water in contact with known hydrophobic and hydrophilic surfaces. Once the network is trained, it can be used to create a spatial map of surface hydrophobicity from the fraction of nearby water molecules that are identified as being in hydrophobic vs. hydrophilic environments.

### 6.4.1 Training methodology

The PointNet is trained from simulations of TIP3P water on hydrophobic and hydrophilic self-assembled monolayer (SAM) surfaces—$CH_3$SAM and OHSAM, respectively. Complete descriptions of the SAM surfaces and system setup are provided in the Appendix C. Example point clouds for training are taken from water molecules wetting the surface. The oxygen atom of a water molecule must be within 0.5 nm of a surface terminal group heavy atom (C or O) to be considered surface-wetting. For each example, the point cloud itself consists of all hydrogen and oxygen atoms within a cutoff distance of 0.6 nm of the central water oxygen. The point clouds include one-hot encoded atomic identity for each atom in the point cloud. Ground truth labels are assigned based upon the surface in the system; point clouds for water molecules on $CH_3$SAM are labeled as examples of hydrophobic environments and point clouds for water molecules on OHSAM are labeled as examples of hydrophilic environments. The point cloud size (i.e., $n$ in Fig. 6.1) was selected in the same manner used for crystal structure identification—two standard deviations above the mean number of points within the cutoff distance (0.6 nm)—and resulted in 82 points per cloud. The training (test) set consisted of a total of 800,000 (200,000) samples split equally between hydrophobic and hydrophilic environments.

### 6.4.2 SAM surfaces

After 50 epochs of training, the cost plateaued and the accuracy on the test set was 84%. The PointNet was trained three times and the reported accuracy is an average of the three trials. It is not particularly surprising that the classification accuracies are much lower than our results for crystal structure identification. Water molecules above both hydrophobic and hydrophilic surfaces are in liquid environments and it seems quite plausible that there is a fair amount of overlap between the distributions of structures sampled in each case. Interestingly, classwise accuracies varied across training trials. The network would achieve ∼90% accuracy on one class but only ∼78% accuracy on the other class. Sometimes the higher-accuracy class was hydrophilic environments while other times the higher accuracy class was hydrophobic environments.

Despite relatively poor accuracy on individual point clouds, we can still extract an average measure of surface hydrophobicity. The bounds of the hydrophobicity scale are determined by the water environments observed near $CH_3$SAM (hydrophobic bound) and OHSAM (hydrophilic bound). To identify numerical bounds for the scale, each surface-wetting water molecule is classified as hydrophobic or hydrophilic with the a pre-trained network. The classification is 'projected' back onto the surface by calculating, for each surface terminal heavy atom, the fraction of surface-wetting water molecules that are classified hydrophobic. This fraction is averaged across terminal groups from the OHSAM and $CH_3$SAM surfaces to generate the bounds for hydrophobic regions (0.95) and hydrophilic regions (0.25).

As a first test of the hydrophobicity scale, we calculate the hydrophobicity map for a SAM surface with alternating stripes of hydrophobic (-$CH_3$) and hydrophilic (-OH) terminal groups. Results are shown in Fig. 6.8(a). The regions identified as hydrophilic (blue surface representation) correspond to OH head groups (blue spheres) and the regions identified as hydrophobic correspond to the locations of -$CH_3$ head groups (red spheres). Intermediate hydrophobicity (white) is found near the boundary between the -OH and -$CH_3$ regions of the surface.

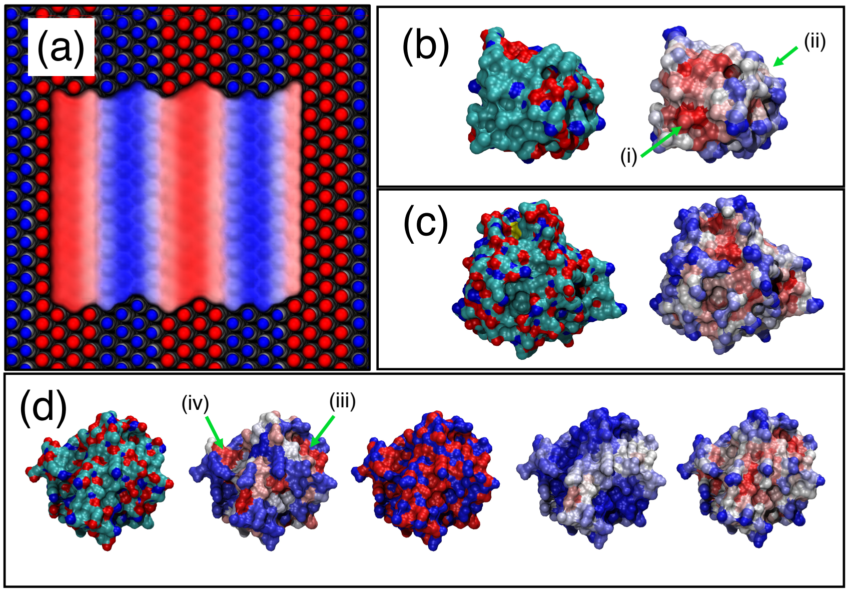

Figure 6.8: Application of PointNet to characterize surface hydrophobicity. (a) SAM surface with alternating -$CH_3$ (red spheres) and -OH (blue spheres) terminal groups. The colored 'surface' representation shows the identification of hydrophobic (red) and hydrophilic (blue) regions by the PointNet. White regions are intermediate hydrophobicity. Predicted surface hydrophobicity of hydrophobin II and CheY is shown in (b) and (c), respectively. Leftmost image shows the protein colored by atom types (red = oxygen, blue = nitrogen, green = carbon) while the rightmost image shows the predicted hydrophobicity. Panel (d) shows a comparison of surface hydrophobicity predictions for a different surface of CheY for our method (far right), Kyte and Doolittle [2] (second from left), Kapcha and Rossky [3] (center), and Shin and Willard [4] (second from right).

### 6.4.3 Protein hydrophobicity

Encouraged by the results on SAM surfaces we tested our method on the surface of two proteins: hydrophobin II (PDB: 2B97) and *Escherichia Coli* CheY (PDB: 3CHY). Complete simulation details are provided in the Appendix C. The same procedure is used to create a spatially resolved surface hydrophobicity map: (1) Water molecules whose oxygen atom is within 0.5 nm of a protein heavy atom are considered surface-wetting. (2) For each surface-wetting water molecule, the point cloud consists of all water atoms within 0.6 nm of the central oxygen. (3) Each point cloud is sent to the trained PointNet to classify the central water molecule as hydrophobic or hydrophilic. (4) The classification is 'projected' back onto the surface by calculating, for each surface terminal heavy atom, the fraction of surface-wetting water molecules that are classified hydrophobic. The bounds of the scale remain the same as used for the striped SAM surface.

The surface hydrophobicities of hydrophobin II and CheY are shown in the rightmost snapshots in Fig. 6.8(b) and Fig. 6.8(c), respectively. The method tends to classify solvent exposed regions as more hydrophilic and buried regions as more hydrophobic. This trend is in agreement with calculations showing that non-polar concave surfaces are more hydrophobic than non-polar convex surfaces [319]. However, the snapshots show that our method does not classify hydrophobicity from geometric considerations alone. Some solvent exposed regions are identified as hydrophobic (e.g., arrow (i)) whereas some are identified as intermediate or hydrophilic (e.g., arrow (ii)). A second view of CheY is shown in Fig. 6.8(d); our method is once again the rightmost snapshot. In panel (d), our results are compared with other methods of quantifying protein surface hydrophobicity. In all cases the more hydrophobic regions are colored darker red and the more hydrophilic regions colored darker blue. Starting second from the left and moving right, the snapshots are the Kyte and Doolittle hydrophobicity scale [2], Kapcha and Rossky atomic hydrophobicity scale [3], and the method of Shin and Willard [4]. For the method of Shin and Willard, the hydrophilic and hydrophobic bounds of the scale are set by the average hydrophobicity of OHSAM and $CH_3$SAM, respectively. There are both similarities and differences across the different methods of quantifying surface hydrophobicity. For example, in panel (d) there is a small region (arrow (iii)) that the Kyte–Doolittle scale (second from left) quantifies as a patch with intermediate to hydrophobic character. All other methods agree that this region is at least intermediate between hydrophilic and hydrophobic, if not showing some hydrophobic character. In contrast, there is another small region (arrow (iv)) that the Kyte–Doolittle scale (second from left) quantifies as a patch with hydrophobic character. The Kapcha–Rossky scale and our method agree with Kyte–Doolitte, but the method of Shin and Willard considers this region hydrophilic. In general, the method of Shin and Willard identifies most of the surface as more hydrophilic than the PointNet—no portion of CheY is determined to be as hydrophobic as $CH_3$SAM. In contrast the PointNet identifies regions that have hydrophobicity roughly equal to $CH_3$SAM.

Multiple methods using water orientations [4] to water density fluctuations [320, 321] have been used to quantify protein surface hydrophobicity. It is difficult to know which method is most correct. Moreover, the precise meaning of surface hydrophobicity may become difficult to quantify within sufficiently buried pockets due to the observer context [318], which found that the surface hydrophobicity map depended on the probe shape. It nonetheless appears that the PointNet method of quantifying surface hydrophobicity provides reasonable results that are in at least partial agreement with other techniques of quantifying surface hydrophobicity on proteins. We view this success as a robust demonstration of the generalizability of the PointNet method for quantifying local structure in molecular simulations.

## 6.5 Conclusions

We introduced a new method to identify local structures in molecular simulations. PointNet, a type of neural network developed for processing point clouds for applications in computer vision, was applied to identify local structure in molecular simulations. Local structure is identified by analyzing point clouds created by the local atomic environment. We tested the method on the problem of crystal structure identification in Lennard-Jonesium, water, and mesophase systems. In all cases, the PointNet approach results in highly accurate classification of crystal structures. The method was demonstrated under realistic use-cases: identifying crystalline seeds in bulk liquid and identifying heterogeneous crystal formation. We also demonstrated that the method generalizes beyond crystal structure identification—the same PointNet approach was shown capable of quantifying local surface hydrophobicity. As a further simple test of the extensibility of the PointNet approach to different types of systems, we trained the network to classify conformations of alanine dipeptide. This small peptide takes two primary configurations in vacuum, which are well separated in the space of backbone $\phi, \psi$ dihedral angles. Using point clouds created from the positions of the backbone atoms relative to the $\alpha$-carbon, the PointNet was able to achieve effectively 100% classification accuracy.

Our work applied the PointNet under novel conditions. As originally developed [300], the PointNet was designed to process large point clouds and extract global features for classification and segmentation. The point clouds consisted of $>$1000 points. We demonstrated that the same network architecture can be used on smaller point clouds and successfully extract subtle differences. We utilized the ability to append features to the coordinates of each point to add the atomic identity of each point while maintaining the invariance to input point order.

The primary strengths of the method are as follows: (1) highly accurate classification of local structures. (2) No system-specific parameterization: only the distance cutoff and maximum number of points in the point cloud must be selected. Our results suggest that, at least for crystal structure identification, 40–80 points results in highly accurate classification. (3) Simple addition of new structures: if at any point it becomes necessary to distinguish between additional structures the network can be retrained.

The PointNet approach should be highly generalizable to a variety of local structures that form in molecular simulations. Possible examples include arrangements of molecular crystals, different polymer configurations, structure in biomolecules, ligand–binding site arrangements, and more. The method will enable rapid structure identification in novel systems that lack order parameters. The PointNet architecture may also prove useful for other examples of machine learning in molecular simulation, e.g., neural network force fields or dimensionality reduction, reaction coordinate identification, and enhanced sampling.

# Chapter 7

# Conclusions and future work

The myriad challenges to studying crystal nucleation in molecular simulations originate with a common cause: the rare event problem. The challenges are thus interdependent. The view advocated by this dissertation is that the solutions will require an equally integrated approach. We address challenges of scale with a big-data-based software framework, challenges of sampling with method development, and challenges of structure identification with machine learning. These specific solutions were motivated by experiences studying crystallization in an exemplar of complex systems rather than a simple model system. However, this dissertation only represents the first iteration of development. The solutions presented here must be applied to complex systems for further iteration and improvement. In addition to extensions and improvements to the individual components, there also remains substantial space for continued integration of the solutions – software with methods, machine learning with methods, and machine learning with software. Continued extension and integration of these solutions will advance the boundaries of simulation studies of crystal nucleation. Below we propose continued avenues of investigation both for improving simulation studies of crystal nucleation and our specific investigations into the nucleation mechanisms of clathrate hydrates.

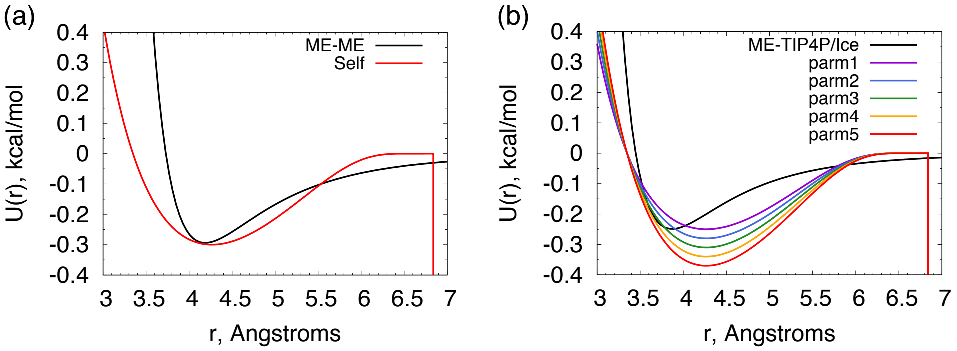

Figure 7.1: (a) Guest–guest and (b) guest–water interaction potentials. OPLS-UA methane (ME) is plotted for reference. $\epsilon_{\text{gw}} = 0.25, 0.28, 0.31, 0.34, 0.37$ kcal mol$^{-1}$ for parm1–parm5, respectively.

## 7.1 Effect of guest solubility on clathrate hydrate nucleation mechanisms and application of contour forward flux sampling to hydrate nucleation

The results presented in Chapter 3 prompt the possibility of different nucleation mechanisms for hydrates formed from guest molecules with different solubility in water. We showed that water structuring played a key role in the nucleation mechanism for the water-soluble ‘XL’ guest molecule. Specifically, order parameters which capture guest ordering (e.g., mutually coordinated guest (MCG) [191]), were poor approximations of the reaction coordinate in comparison to order parameters that more explicitly capture water structure (e.g., dihedral order parameter (DHOP) [131], Baez and Clancy order parameter [192]). In contrast, a study of the sparingly soluble ‘M’ guest molecule found that the MCG order parameter was a good approximation of the reaction coordinate [203]. These findings present a discrepancy and warrant further investigation.

There are explanations beyond water solubility for the apparent discrepancy. The M guest is smaller in size than XL guest. It also forms structure I (sI) hydrate whereas XL forms structure II (sII) hydrate. Due to it’s larger size, the XL guest is unable to occupy the $5^{12}$ cages of sII hydrate. Several studies [94, 27, 204, 101, 104] report the formation of $5^{12}$ cages in the early stages of hydrate nucleation. Since the XL guest is unable to occupy and stabilize this cage type, it is possible that water structuring appears as a key step for XL hydrate because of the necessity of forming empty $5^{12}$ cages.

To disentangle these conflicting possibilities we developed a series of guest molecules (parm1–parm5) that are the same size but have varying water solubility. We use the coarse-grained mW water model [174] as this allows us to isolate the effects of guest solubility. The mW model also has faster hydrate growth and dissociation dynamics compared with all atom models (e.g., TIP4P/Ice). The latter point substantially decreases the computational cost of the calculations. Guest molecules interact with each other and mW water through the two-body term of the Stillinger-Weber (SW) potential [185] via a parameter that controls the size ($\sigma$) and a parameter that controls the interaction energy ($\epsilon$) [95]. The guest–guest interaction parameters ($\sigma_{\mathrm{gg}} = 3.8$ Å,$\epsilon_{\mathrm{gg}} = 0.3$ kcal mol$^{-1}$) are held constant for all five guest molecules. The effective size of guest–water interactions is also held constant ($\sigma_{\mathrm{gw}} = 3.8$ Å). The guest–water interaction energy ($\epsilon_{\mathrm{gw}}$) is systematically varied from parm1–parm5. Guest–guest and guest–water potentials are plotted for parm1–parm5 in Fig. 7.1(a) and Fig. 7.1(b), respectively. All parameters were selected to fall within the sI forming region of the phase diagram [95]. The solubility of the parm1–parm5 guest molecules in water and melting temperatures of clathrates formed from parm1–parm5 are reported in Fig. 7.2. Solubilities were calculated from simulations of a guest-rich fluid phase in coexistence with a water-rich liquid phase at 100 bar and temperatures from 280 to 380 K. The hydrate melting temperatures at 100 bar were estimated with the direct coexistence method [322]. For each parameter set it was confirmed that sI with all cages occupied was the most stable hydrate crystal (i.e., had the highest melting temperature). Other crystals tested were sI with $5^{12}$ cages occupied, sI with $5^{12}6^{2}$ cages occupied, sII with $5^{12}$ cages occupied, sII with $5^{12}6^{4}$ cages occupied, and sII with all cages occupied.

In Chapter 5 we demonstrated the benefits of contour forward flux sampling (cFFS) when the optimal order parameter (i.e., reaction coordinate) is not known *a priori*. The hydrate systems described above offer an ideal opportunity to apply cFFS to study crystallization in a system where the optimal order parameter is unknown. Based upon our findings in Chapter 3 as well as the work of others [203], we hypothesize that the reaction coordinate for hydrate nucleation will shift from an emphasis on guest ordering to water structuring with increasing guest solubility. However, we do not know if this change will be sharp or continuous with guest solubility, or the level of guest solubility required to observe the crossover. It is also possible that no such crossover will occur. As discussed previously, there are other differences between the soluble XL guest and sparingly soluble M guest which could explain the relative importance of water structuring in XL nucleation. Furthermore, the effectiveness of our water structuring order parameter, DHOP, has never been tested for hydrate

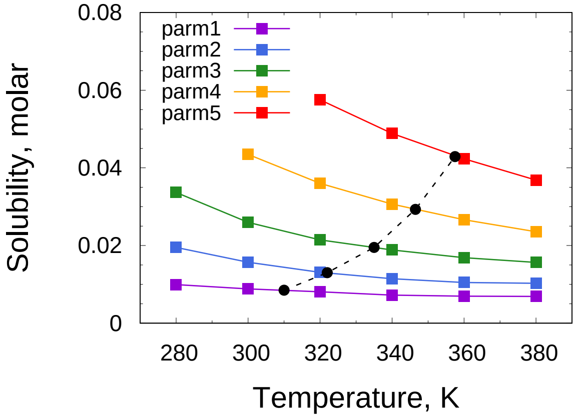

Figure 7.2: Guest solubility (filled squares) as a function of temperature at 100 bar. The melting temperature of sI hydrate with all cages occupied at 100 bar is reported as filled circles. Lines are to guide the eye.

nucleation of sparingly soluble guest molecules. It is possible that DHOP is a good order parameter for hydrates formed from both soluble and sparingly soluble guests – i.e., there may be degeneracy of good order parameters at low guest solubility. Given the uncertainties regarding the optimal order parameter, we will perform cFFS with two order parameters, DHOP and MCG.

The conditions for our studies are selected to maintain constant supercooling across the different guest molecules in order to isolate solubility effects. It is also necessary to strike a balance between supercooling, critical nucleus size, and nucleation rate. Ideally our studies would be performed at relatively small supercooling. However, the nucleation rate decreases and the critical nucleus size increases with decreasing supercooling (i.e., higher temperature). As the critical nucleus size becomes larger, the system sizes required to prevent system size effects increase. Likewise, as the nucleation rate becomes smaller, the rarity of nucleation increases and the studies will require longer simulations/more computational effort. During our solubility studies, we observed nucleation in straightforward MD simulation in <100 ns for parm4 at 280 K and parm5 at 300 K. These temperatures represent $0.81T_\mathrm{m}$ and $0.84T_\mathrm{m}$, respectively. The results provide order-of-magnitude estimates of the nucleation rate as $10^{32}$ events $\mathrm{m}^{-3}\mathrm{s}^{-1}$ and the critical nucleus size ∼100 water molecules. Previous FFS studies of soluble [104] (at $0.72T_\mathrm{m}$) and sparingly soluble [108] (at $0.83T_\mathrm{m}$) guests resulted in critical nuclei of ∼180 and ∼350 water molecules, respectively. A seeding study of the M guest at $0.92T_\mathrm{m}$ found a critical nucleus size of ∼2000 water molecules [62]. These results provide rough bounds on expected critical nucleus sizes and nucleation rates. We select the initial conditions of our studies as $0.88T_\mathrm{m}$.

## 7.2 SAFFIRE: Release code, add cFFS, and extend to other methods

SAFFIRE enabled the FFS calculations reported in Chapter 3, which comprised over 500,000 individual MD simulations and required 33 days on a cluster of 30 × 16-core Intel Xeon E5-2665 CPUs. The code should be documented and released so that it can be used by other groups. The SAFFIRE framework – using Hadoop streaming [208] to manage the execution of simulation/analysis tasks and then parsing the results and making decisions with Cascading [231] or Spark [323] can also be applied to other advanced sampling techniques. As a starting point, adding cFFS to SAFFIRE will increase the accessibility of the method and make it easier to apply for studies of complex systems (e.g., hydrate nucleation). If the constrained forward shooting TIS method is efficient for studies of crystal nucleation (see Sec. 7.4) then it should be implemented in SAFFIRE as well. Python-based softwares have recently been released for transition path sampling and transition interface sampling methods [147, 148, 149]. Integrating SAFFIRE (for resource management) with these packages (for algorithm management) may prove valuable, but likely represents a challenging and substantial undertaking.

## 7.3 Developing FFS methods that are capable of screening larger numbers of order parameters simultaneously

cFFS represents one approach to perform FFS with multiple order parameters. Unfortunately, cFFS is currently limited to 2–3 order parameters. One possibility is to combine cFFS with machine learning and/or dimensionality reduction techniques to screen through a larger number of order parameters but reduce those to 2 or 3 collective coordinates. This would restrain cFFS to a manageable number of dimensions. A discussion of the challenges to incorporating these types of approaches with cFFS is provided in Chapter 5.

An entirely different approach to performing FFS with multiple order parameters would be to develop a method whereby several FFS simulations with different order parameters were performed as parallel replicas. With the proper acceptance/rejection scheme, trajectories could be swapped between different replicas in order to enhance sampling along different order parameters, allowing the system to relax in orthogonal directions and navigate around orthogonal barriers. Compared with cFFS this type of approach would scale much better to larger numbers of order parameters because it would not suffer from the curse of dimensionality. The challenge is designing the appropriate swapping rules to maintain the applicability of FFS to nonequilibrium systems. Swapping trajectories between multiple parallel replicas would be easier to implement with a transition interface sampling (TIS) approach where acceptance/rejection criteria can be developed from knowledge of the correct phase space distribution. However, this would require sacrificing applicability to nonequilibrium systems. Furthermore, TIS is already much less sensitive than FFS to the choice of order parameter, reducing if not eliminating the benefits of such an approach.

## 7.4 Testing the efficiency of constrained forward shooting transition interface sampling for crystal nucleation

cFFS attempts to address one challenge of FFS – selecting a good order parameter *a priori*. However, even with a good order parameter, FFS/cFFS will remain computationally costly for studies of crystal nucleation in complex systems. In ice and hydrate systems, the most computationally expensive portion of the FFS calculations is waiting for trajectories near the critical nucleus size to either grow, and reach the next interface, or dissociate, and return to state $A$. This portion of the computational cost is unrelated to the rarity of the event but rather the slow diffusivity in the nucleus size coordinate. In an FFS calculation that required 22 million CPU-hours, Haji-Akbari and Debenedetti reported that the average time for a trajectory to return to $A$ was nearly an order of magnitude longer than the time required to reach the next interface [53]. Worse yet, the trajectories returning to $A$ are discarded in FFS. Therefore, the large majority of computational effort is 'wasted' on trajectories which ultimately do not enhance the sampling of the transition path ensemble.

One possibility is to make use of trajectories returning to $A$ through a time reversal move. This requires sacrificing applicability to nonequilibrium systems and requires stochastic dynamics. However, the benefits are potentially substantial for equilibrium systems with slow diffusivity along the order parameter. Bolhuis described just such an idea with a constrained forward shooting replica exchange transition interface sampling (TIS) method [324]. To keep the algorithm easier to parallelize, I propose using constrained forward shooting TIS without replica exchange between interface ensembles. Imagine a trajectory initiated from some $\lambda_i$ that eventually returns to $A$. If a time reversal move were applied to that trajectory, the time reversed trajectory would be guaranteed to contain a (different) first crossing of $\lambda_i$. Shooting from this new first crossing point would enhance sampling at $\lambda_i$ and prevent the backwards portion of the trajectory from going to waste.

I will use Figure 7.3 to further illustrate the idea. The combination of the time reversal move, constraining the shooting points to the interface, and only shooting in the forward direction results in no rejected shooting moves. Imagine some valid first crossing phase point at $\lambda_i$, shown as the red open circle. The path history is shown as the red dashed line. Borrowing notation from Chapter 1, a valid first crossing phase point is one that satisfies the criteria $h_{\Omega^{[i]+}}(x) = 1$. Forward shooting is used to generate a new trajectory (solid red line) from the red open circle. The trajectory is terminated when it reaches $A$ or $B$. In this example, the trajectory returns to $A$. A time reversal move would create the blue trajectory. The acceptance probability for this move is unity so long as the original (red) trajectory terminates in $A$ rather than $B$. The first crossing phase point at $\lambda_i$ for the time reversed trajectory is shown by the blue open circle. A forward shooting move then generates a new trajectory from the blue phase point (dark green path). Note that stochastic dynamics is required to ensure trajectory divergence. The green trajectory crosses $\lambda_{i+1}$ before returning to $A$. A time reversal move applied to the green trajectory would create the light yellow trajectory. This move has provided two new first crossing phase points; one at $\lambda_i$ and one at $\lambda_{i+1}$. The point at $\lambda_i$ can be used to continue sampling the $\Omega^{[i]+}$ ensemble. The point at $\lambda_{i+1}$ can be saved and used at a later stage to initiate sampling of the $\Omega^{[i+1]+}$ ensemble. A possible algorithm follows:

1. Run basin simulations. Place $\lambda_\mathrm{A}$ and $\lambda_0$. Calculate the flux from $\lambda_\mathrm{A}$ to $\lambda_0$.
2. For `i=0:i<N:i++`:
    (a) Harvest initial paths for the $\Omega^{[i]+}$ ensemble. These are paths that begin in $A$, cross $\lambda_i$ at least once, and terminate in $A$ or $B$.

    (b) Randomly select a path in the $\Omega^{[i]+}$ ensemble.

    (c) Attempt a shooting move or a time reversal move with 50% probability assigned to each.
    - Shooting move: Initiate a single trajectory from the first crossing phase point at $\lambda_i$. Continue the trajectory until it reaches $\lambda_\mathrm{A}$ or $\lambda_\mathrm{B}$. The move is always accepted.
    - Time reversal move: Reverse the ordering of the time slices for the selected trajectory
        - Accept if new trajectory originates in A.

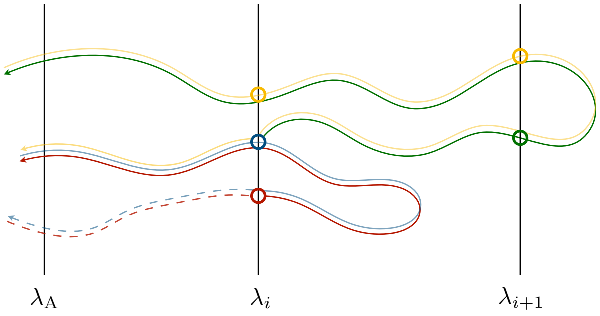

Figure 7.3: Schematic of constrained forward shooting TIS. Open circles represent first crossing phase points. Solid red and green lines represent trajectories are generated through shooting. Transparent blue and yellow lines represent trajectories are generated with a time reversal move.

- Reject otherwise. Recount original trajectory in the $\Omega^{[i]+}$ ensemble.

(d) Return to Step 2b. Repeat until $\mathcal{P}(\lambda|\lambda_i)$ converges. The convergence of distributions in orthogonal coordinates, $\mathcal{P}(\lambda^{\perp}|\lambda)$, can also be used to confirm convergence of the $\Omega^{[i]+}$ path ensemble.

(e) Continue to `i+1`

The final rate constant can be calculated from the same expression used for TIS and FFS: $k_{AB} = \Phi_{A0} \prod_{i=0}^{N-1} \mathcal{P}(\lambda_{i+1}|\lambda_i)$. Alternatively, the weighted histogram analysis method can also be used to tie together $\mathcal{P}(\lambda|\lambda_i)$ calculated for each path ensemble [324]. The algorithm can be parallelized by performing Step 2b for many paths at once.

Strictly speaking, stochastic dynamics are required in order for the acceptance probability of the shooting move to be unity. In the absence of stochastic dynamics, perturbation of the shooting point is required for trajectory divergence. If the shooting point is perturbed, backwards integration is required to confirm that the modified shooting point indeed satisfies the $h_{\Omega^{[i]+}}(x) = 1$ criteria. If the criteria is not met, the shooting move must be rejected. In other words, deterministic dynamics would largely defeat the primary benefit of the constrained forward shooting method — guaranteed acceptance of all shooting moves. In the case of crystal nucleation there are a few possibilities (1) use weak coupling to a stochastic (e.g., Langevin) thermostat, (2) Lyapunov instability may provide sufficient stochasticity for trajectory divergence or (3) stochasticity along the nucleus size coordinate may be sufficiently large that drawing new momenta at the shooting point is acceptable.

Compared to conventional TIS/replica exchange TIS, the benefits are (1) no rejected shooting moves, and (2) a somewhat simplified algorithm that is easy to parallelize. Compared to FFS, the benefits are (1) backward paths can contribute to sampling with the time reversal move, and (2) each $\Omega^{[i]+}$ path ensemble is allowed to relax in directions orthogonal to the order parameter, $\lambda$. This latter point should result in less dependence on the choice of order parameter compared with FFS.

I propose to test this method on ice nucleation of mW water. Ice nucleation of mW water is computationally approachable compared with clathrate hydrate nucleation or all-atom ice nucleation. Nonetheless, the system is sufficiently complex that crystal nucleation of mW water displays some similar characteristics to crystal nucleation in those more complex systems. This test case will allow comparison between the computational cost of constrained forward shooting TIS and FFS. It also provides a system to test the above listed possibilities for handling stochasticity.

## 7.5 Adding segmentation capabilities to the PointNet approach to local structure identification

The PointNet approach to local structure identification presented in Chapter 6 appears promising. However, improvements could reduce the computational cost of the method and bring the method closer to our vision – using machine learning models to directly process raw output of molecular simulations. These two aspects are related. For the purposes of explanation, imagine we are using the PointNet to classify the crystal structure of each atom in an atomic system with $N$ atoms of a single atom type. In the current approach, $N$ point clouds would be generated for a single frame of the simulation. Each point cloud would describe the relative positions ($x_i$,$y_i$,$z_i$) of all atoms within some cutoff distance of a central atom. All $N$ point clouds are sent through a trained PointNet, which returns a label (i.e., crystal type) for each point cloud. The label predicted by the PointNet is then applied to the central atom of each point cloud. Thus, $N$ samples must be sent through the network for each frame of the simulation. Furthermore, the point clouds must be preprocessed. From the raw simulation output, we identify neighbors (i.e. atoms within the cutoff distance) for each atom, and then translate the positions of the central atom and all neighbors such that the central atom is located at (0,0,0).

There are good reasons for the current strategy: (1) symmetry with respect to global translation is addressed because the central atom for each point cloud is always at (0,0,0), (2) symmetry with respect to rotation is easily achieved by applying a random rotation to each point cloud prior to sending it through the PointNet, (3) any periodic boundary conditions can be applied when preprocessing the point clouds; the PointNet doesn't need to understand or account for PBCs, and (4) the system sizes (i.e., number of atoms) do not need to be the same in the systems used for training/testing/production since each sample sent through the PointNet describes only the environment surrounding some central atom and has a fixed number of points.

One option to process all $N$ atoms in the system at once is to use segmentation rather than classification. In segmentation, a single frame with all $N$ atoms would be treated as a single point cloud. The PointNet would return a prediction of the crystal structure for each atom in the point cloud. In this way, the point cloud is 'segmented' rather than 'classified'. However, traditional segmentation may be challenging to apply in molecular systems. This arises from the way segmentation works. Segmentation first calculates an overall feature vector for the entire point cloud. Then, point-specific features are appended and a point-specific classification is determined from the combination of the overall feature vector and point-specific information.

In molecular systems, the environment of a specific atom is determined by local rather than global factors. In other words, the structural environment of atom $\mathrm{a}_i$ is unaffected by some atom $\mathrm{a}_j$ if $\mathrm{a}_j$ is sufficiently spatially removed from $\mathrm{a}_i$. Imagine a situation where a large portion of the system is one phase (e.g., liquid), and a small portion is another phase (e.g., crystal embryo). The global feature vector for such a system would probably be very similar to the global feature vector of a pure liquid system. It is far from certain that the point-specific features of atoms in the crystal embryo would be sufficiently strong to affect the final segmentation outcome. This problem would be exacerbated by challenges training such a network. In the current approach, training examples are generated from simulations of pure phases. A ground-truth label is thus easy to apply for a point cloud extracted from a simulation of a given phase. Unfortunately, pure phases would probably be insufficient to train the PointNet for segmentation. If every atom in a training example belongs to the same phase, the global feature vector encodes all the information required to predict the phase of each individual atom (i.e., they are all the same phase). The network thus never learns to combine information from the global feature vector with point-wise information to create a point-wise classification. Thus, training the PointNet for segmentation will require simulations with phase coexistence, in which case it is unclear how the ground truth labels for atoms near the phase boundary will be determined, or, it will require a creative method of generating artificial training examples with multiple phases present. A few other challenges to using segmentation will include:

- How to handle PBCs. One option is to append box dimensions to the input data. A more promising strategy is padding. Points near the edges of the simulation box would be duplicated in their periodic location just outside the simulation box. Unfortunately this requires some preprocessing and increases the size of the point cloud.
- How to handle systems with different numbers of atoms. The zero-padding approach used in our existing work may work in the case of segmentation as well.
- How to handle sensitivity to the position and orientation of crystal phases in the simulation box. This is currently handled by translating each point cloud to (0,0,0) and applying a random rotation to each training example.

There are clearly a range of challenges to using segmentation with the PointNet in molecular systems. Nonetheless, it would substantially improve the method by reducing the computational cost and completely eliminating trajectory preprocessing.

# Appendices

# Appendix A Supporting Information for Chapter 3

## A.1 Order parameters

### A.1.1 RNGOP

RNGOP is entirely based on the types of rings that each water molecule participates in. The procedure is as follows:

1. Identify all the unique 5-membered rings and 6-membered rings of water molecules in the system where all 5- and 6- membered rings must be simple cycles [325].

2. For each water molecule, count the number of 5-membered rings and 6-membered rings that it participates in.

3. Tag a water molecule as hydrate-like if it meets any of the following criteria:

   - Belongs to four 5-membered rings and two 6-membered rings

   - Belongs to five 5-membered rings and one 6-membered ring

   - Belongs to six 5-membered rings and zero 6-membered rings

4. Identify the largest cluster of hydrate-like water molecules, where two water molecules must be first neighbors to belong to the same cluster.

## A.2 Forward flux sampling

Table A.1: Summary of basin simulations

| Temperature, K | $\lambda_A$ | $\lambda_0$ | $N_{cross}$ | $V$, nm$^3$ | Total Time, ns | $\Phi_0$, m$^{-3}$s$^{-1}$ |
|---|---|---|---|---|---|---|
| 230 | 19 | 41 | 31960 | 252.0 | 487.7 | $2.60 \times 10^{35}$ |

Details of the basin simulations are reported in Tab. A.1. Details of the FFS, including interface values, $\lambda_i$, the number of configurations harvested at each interface, $N_{conf}$, the total number of trajectories initiated from each interface, $N_{traj}$, the number of trajectories that successfully crossed the next interface, $N_{cross}$, the number of trajectories that returned to basin, $N_{basin}$, and the probability of advancing to the next interface, $P(\lambda_{i+1}|\lambda_i)$, are reported in Table A.2.

Table A.2: Forward flux sampling details

| $i$ | $\lambda_i$ | $N_{conf}$ | $N_{traj}$ | $N_{cross}$ | $N_{basin}$ | $P(\lambda_{i+1}|\lambda_i)$ |
|---|---|---|---|---|---|---|
| 0 | 41 | 778 | 40000 | 2860 | 37140 | 0.071500 |
| 1 | 47 | 377 | 40000 | 7207 | 32793 | 0.180175 |
| 2 | 60 | 1112 | 40000 | 12052 | 27948 | 0.301300 |
| 3 | 76 | 1775 | 40000 | 20564 | 19436 | 0.514088 |
| 4 | 91 | 3015 | 20000 | 10386 | 9614 | 0.519300 |
| 5 | 116 | 1421 | 20000 | 13590 | 6410 | 0.679500 |
| 6 | 137 | 1838 | 20000 | 14969 | 5031 | 0.748450 |
| 7 | 179 | 1807 | 10000 | 9577 | 423 | 0.957700 |
| 8 | 236 | 1064 | 10000 | 9985 | 15 | 0.998500 |
| 9 | 286 | 1101 | 10000 | 9998 | 2 | 0.999800 |
| 10 | 339 | N/A | N/A | N/A | N/A | N/A |

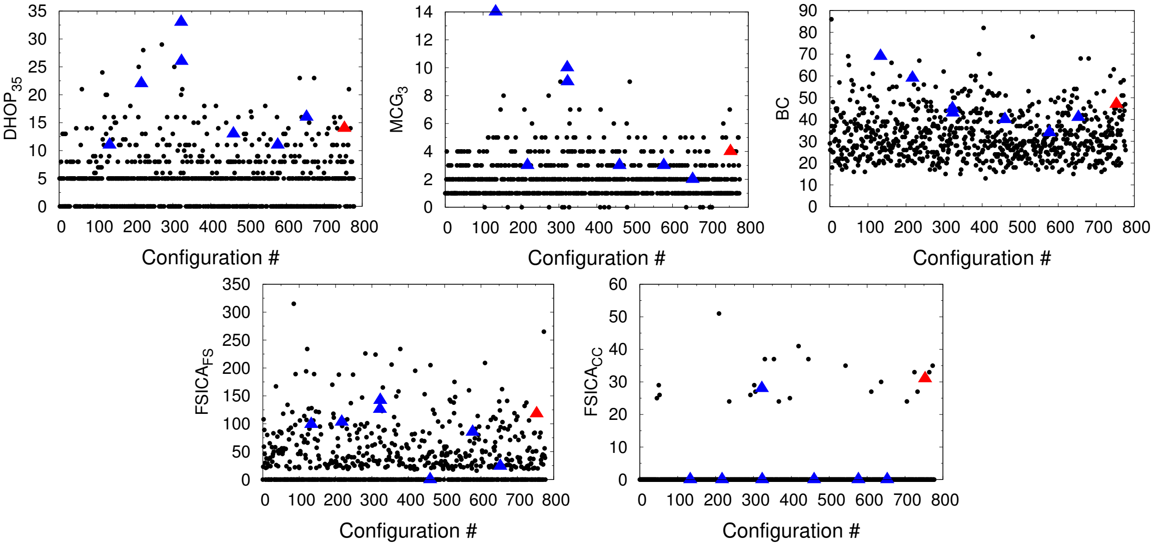

Figure A.1: Value of several order parameters for all 778 configurations at $\lambda_0$. C753 is indicated with a red triangle. The 7 other configurations which spawned at least one transition path are shown as blue triangles. All other configurations at $\lambda_0$ are shown with black points.

## A.3 C753

We found that 93% of all successful transition pathways from FFS originated from a single configuration at $\lambda_0$ (C753). The remaining transition pathways originated from 7 other configurations at $\lambda_0$. To show that C753 is not an outlier, the values of several order parameters for all configurations at $\lambda_0$ are reported in Fig. A.1. Though C753 has a higher than average value for many order parameters, C753 does not fall outside the space sampled by many other configurations at $\lambda_0$. Even in the case of FSICA$_{\mathrm{CC}}$ (bottom right panel), there are 23 other configurations with similarly-sized or larger complete cage clusters. Of these 23 other configurations with complete cage clusters, only 1 other configuration (C322) spawned any transition paths. Furthermore, C753 does not have a particularly high value of most order parameters when compared with the 7 other configurations which spawned the remaining 7% of successful transition pathways.

Snapshots of the FSICA complete cage clusters for C322 and C753 (the two configurations at $\lambda_0$ with a non-zero FSICA$_{\mathrm{CC}}$ cluster with spawn successful transition paths) are shown in Fig. A.2(a)-(b). Two other representative configurations from $\lambda_0$ with non-zero FSICA$_{\mathrm{CC}}$ clusters that *do not* spawn any successful transition paths are shown in Fig. A.2(c)-(d).

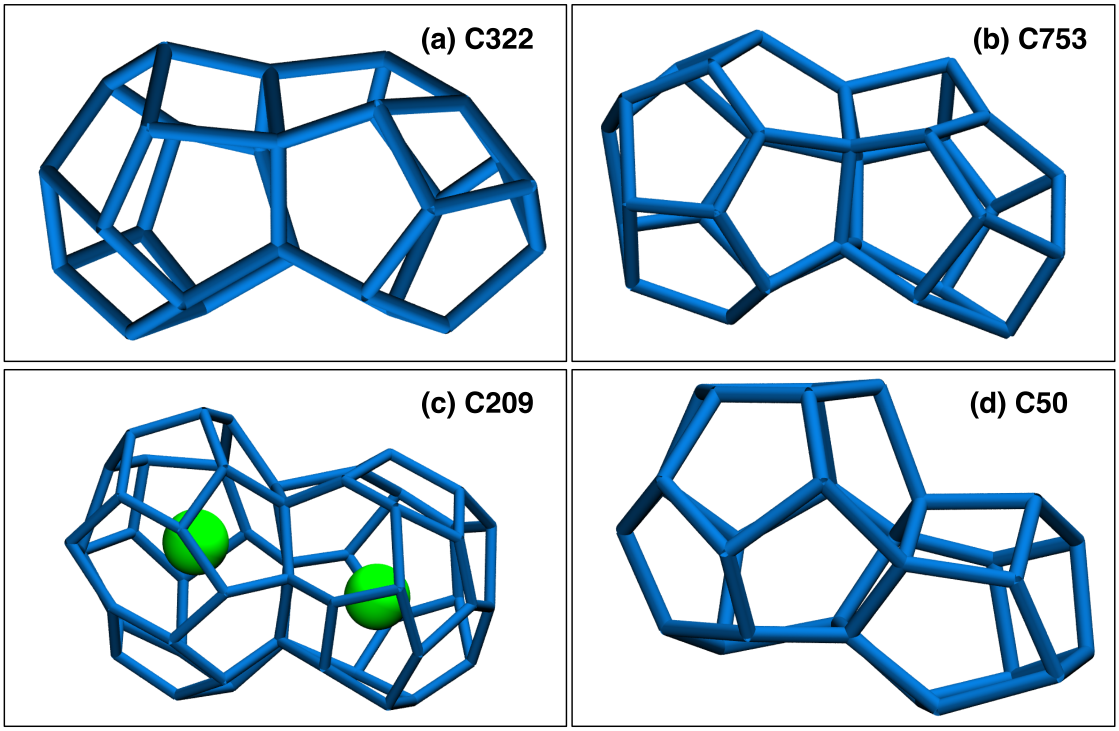

Figure A.2: Snapshots of the FSICA complete cage cluster for four configurations from $\lambda_0$. Water molecules are shown as blue bonds and guest molecules which occupy a cage are shown as green spheres. (a) and (b) are the two configurations at $\lambda_0$ which spawned transition pathways and have non-zero $\mathrm{FSICA}_{\mathrm{CC}}$ cluster sizes. (c) and (d) are two representative configurations from $\lambda_0$ which did not spawn any successful transition pathways, but have a non-zero $\mathrm{FSICA}_{\mathrm{CC}}$ cluster size. (a) consists of a $4^2 5^8$ cage face-sharing through a 5-membered ring (5MR) with a $4^3 5^8$ cage. (b) consists of a $5^{12}$ cage face-sharing through a 5MR to a $4^2 5^8$ cage. (c) consists of a $4^1 5^{12} 6^4$ cage face-sharing through a 6MR to a $4^1 5^{10} 6^5$ cage. (d) consists of a $5^{12}$ cage face-sharing through a 5MR to a $4^3 5^6$ cage.

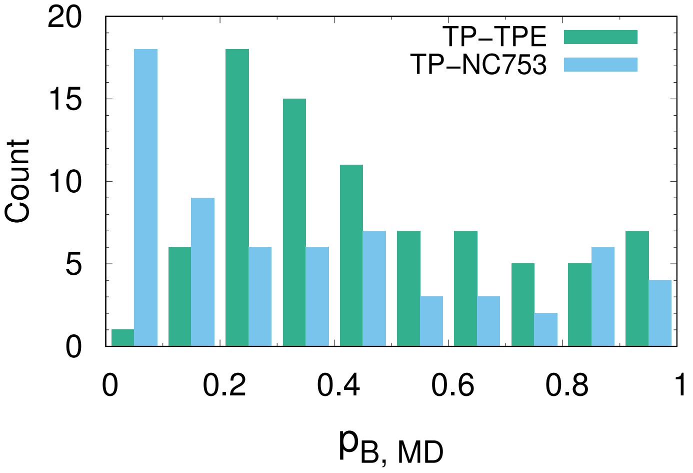

Figure A.3: Histogram of $p_{\mathrm{B,MD}}(\mathbf{x})$ values for configurations in TP-TPE and TP-NC753.

## A.4 Histogram of committor probabilities

A histogram of the $p_{\mathrm{B,MD}}(\mathbf{x})$ values for TP-TPE and TP-NC753 is provided in Fig. A.3. Despite attempting to select configurations with an evenly distributed $p_{\mathrm{B,MD}}(\mathbf{x})$, both TP-TPE and TP-NC753 have more configurations with $p_{\mathrm{B,MD}}(\mathbf{x}) < 0.5$ than $p_{\mathrm{B,MD}}(\mathbf{x}) > 0.5$. Because $p_{\mathrm{B}}$ is non-linear near $p_{\mathrm{B}} = 0.0$ and $p_{\mathrm{B}} = 1.0$ we further evaluated the sensitivity of our analysis by fitting single OP models with only configurations with $0.1 < p_{\mathrm{B,MD}}(\mathbf{x}) < 0.9$. We find that our key results and the general rankings for TP-TPE and TP-NC753 are unchanged.

## A.5 Fits of single OP models to TP-TPE

Graphs of the fits of single OP models to $p_{\mathrm{B,MD}}(\mathbf{x})$ are provided in Fig. A.4.

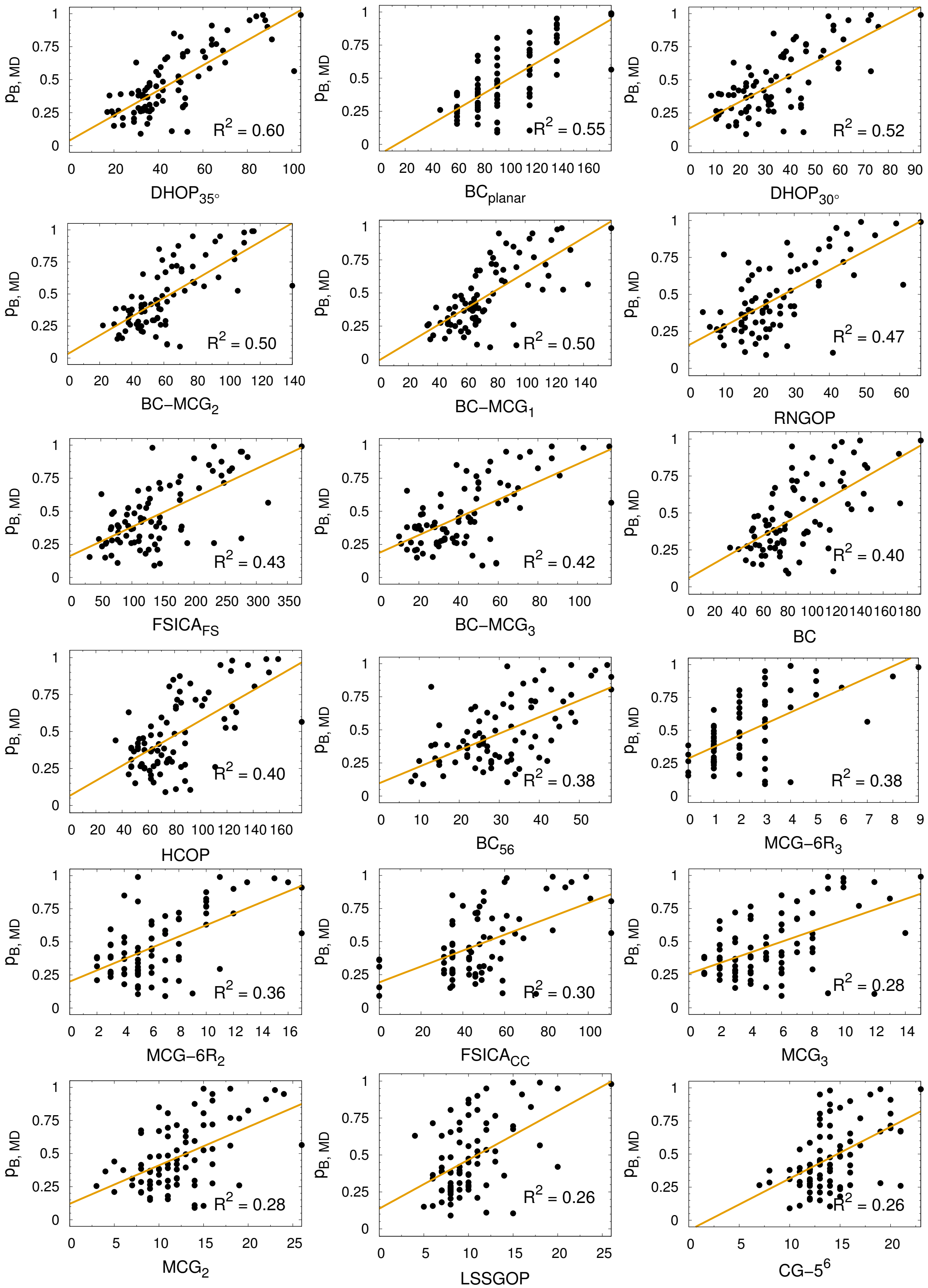

Figure A.4: Linear fits of the top 18 OPs vs. $p_{\mathrm{B,MD}}(\mathbf{x})$ for TP-TPE.

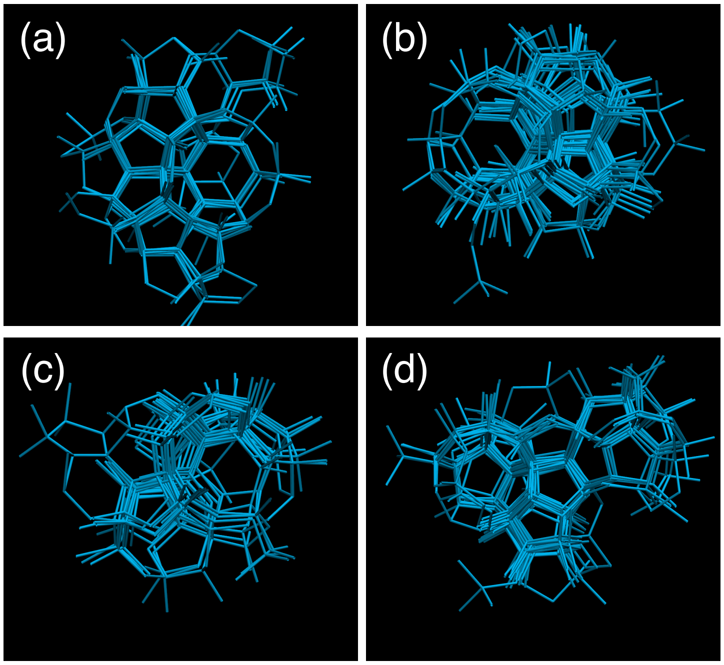

Figure A.5: Snapshots from four independent nucleation trajectories generated with straightforward MD simulations. Water molecules belonging to the largest cluster of DHOP$_{35^\circ}$ are shown with cyan bonds. The snapshots are overlaid frames from ∼100-300 ps as the trajectory crosses through the transition state as characterized by the size of DHOP$_{35^\circ}$.

## A.6 Straightforward MD nucleation simulations

Twenty straightforward MD simulations were performed with the same system and at the same conditions as the FFS calculations. All trajectories nucleated within 600 ns. Representative snapshots from four trajectories are shown in Fig. A.5.

# Appendix B Supporting Information for Chapter 5

## B.1 Details of Langevin dynamics and FFS/cFFS parameters

The behavior of a particle on the four potential energy surfaces (PESs) was described by the Langevin equation, $\ddot{\boldsymbol{q}} = -\nabla U(\boldsymbol{q}) - \gamma\dot{\boldsymbol{q}} + \sqrt{2\gamma k_B T}R(t)$, where $\boldsymbol{q}$ represents the coordinates of the particle, $U(\boldsymbol{q})$ is the PES, $\gamma$ is the friction coefficient, $k_B$ is the Boltzmann constant, $T$ is the reduced temperature, and $R(t)$ is delta-correlated Gaussian random noise with zero mean and unit variance. The dynamics were generated with the velocity Verlet integrator with a time step of 0.01. Simulations were performed at $\gamma = 5.0$. We confirmed that $\gamma = 5.0$ provides sufficient stochasticity for FFS by comparing rates estimated from straightforward Langevin dynamics (SLD) with rates estimated from FFS for a range of $\gamma$ from 0.01 to 100 at $\beta = 2.5$ (data not reported). FFS rates agreed with SLD for all surfaces at $\gamma \geq 1.0$. At $\gamma < 1.0$ FFS underestimated the rate constant. It is possible that $\gamma = 5.0$ does not provide sufficient stochasticity at $\beta = 5.0$. This may explain why $\text{FFS}_{\text{opt}}$ and cFFS rates at $\beta = 5.0$ agree more closely with SLD for surfaces with a more flatter and thus more diffusive transition region (PES-1 and PES-4).

SLD results were averaged over 50 and 400-600 independent simulations of length $10^7$ time units at $\beta = 2.5$ and $\beta = 5.0$, respectively. FFS/cFFS results were averaged from three independent trials. At $\beta = 2.5$, FFS/cFFS was performed with 10,000 trajectories per interface at 1,000 configurations per interface. At $\beta = 5.0$, FFS/cFFS was performed with 40,000 trajectories per interface and 4,000 configurations per interface.

## B.2 Potential energy surfaces

Equations for the PESs used in this work are reported in Eqns. 1–4. Barrier heights are provided in Table B.1. For each surface, the negative-$x$ minimum is considered state $A$ and the positive-$x$ minimum state $B$. The potential energy difference from the minimum of $A$ to the lowest potential energy transition state is 3.4 for all surfaces. PES-1 and PES-2 have a single transition tube, while PES-3 and PES-4 have two transition tubes.

$$\begin{aligned} V_{\text{PES-1}}(x,y) =& 0.02(x^4+y^4) - 4\exp(-((x+2)^2+(y+2)^2)) \\ &- 4\exp(-((x-2)^2+(y-2)^2)) + 0.3(x-y)^2 + 0.0026 \end{aligned} \tag{1}$$

Table B.1: Barrier heights for PESs. TS1 and TS2 are the positive-$y$ and negative-$y$ transition states, respectively.

| | $\mathrm{A}\xrightarrow{\mathrm{TS1}}\mathrm{B}$ | $\mathrm{A}\xrightarrow{\mathrm{TS2}}\mathrm{B}$ |
|---|---|---|
| PES-1 | 3.402 | N/A |
| PES-2 | 3.403 | N/A |
| PES-3 | 3.403 | 3.403 |
| PES-4 | 3.405 | 4.143 |

$$\begin{aligned} V_{\text{PES-2}}(x,y) =& 0.02(x^2+y^2)^2 - 5.196\exp(-0.08(x+3.5)^2 - 1.5(y+1.3)^2) \\ & - 5.196\exp(-0.08(x-3.5)^2 - 1.5(y-1.3)^2) + 0.30914 \end{aligned} \tag{2}$$

$$\begin{aligned} V_{\text{PES-3}}(x,y) =& 0.02(x^4+y^4) - 3.73\exp(-((x+2)^2/8 + (y+2)^2/8)) \\ & - 3.73\exp(-((x-2)^2/8 + (y-2)^2/8)) \\ & + 3\exp(-(x^2/2 + y^2/15)) + 2\exp(-(x^2/2 + y^2/2)) - 0.5085 \end{aligned} \tag{3}$$

$$\begin{aligned} V_{\text{PES-4}}(x,y) =& - 2.93\exp(-((x-3)^2/2 + (y-2.5)^2/2)) \\ & - 2.93\exp(-((x+2)^2/2 + (y+2)^2/2)) \\ & + 3\exp(-0.32((x+1)^2 + (y-2)^2 + 12(x+y-2.7)^2 - 1)) \\ & + 6\exp(-0.15((x-2)^2 + (y-1)^2 + 10(x+y)^2 - 1)) \\ & + 0.005(x^4+y^4) - 0.627 \end{aligned} \tag{4}$$

## B.3 TPE sampling at $\beta = 2.5$

Transition path ensemble (TPE) sampling at $\beta = 2.5$ is reported in Fig. B.1. All methods result in similar sampling to the SLD results reported in Fig. 1 of the main text. All methods successfully sample both transition tubes for PES-3 and PES-4 at this higher temperature.

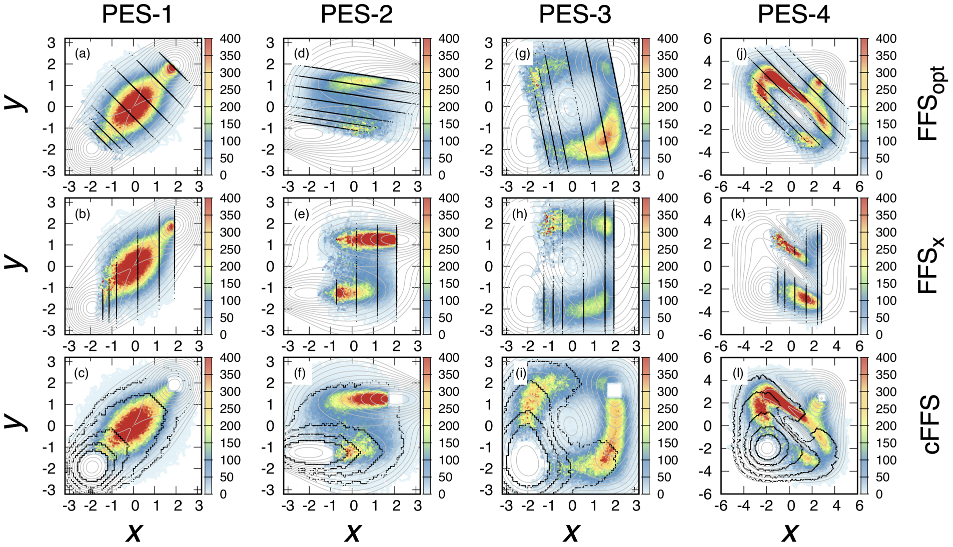

Figure B.1: Comparison of interface placement and TPE sampling generated with FFS$_{\mathrm{opt}}$, FFS$_{\mathrm{x}}$, and cFFS on PES-1 – PES-4 at $\beta = 2.5$. PES contours are shown as gray lines. Configurations at each interface are shown with black points. TPE sampling is represented by the heat map.

## B.4 FFS sampling on PES-3 at $\beta = 5.0$

FFS struggles to sample both transition tubes at $\beta = 5.0$ on PES-3. TPE sampling for all three runs of FFS with three different order parameters and cFFS are reported in Fig. B.2. Even with the optimal order parameter $(5x + y)$, FFS fails to equally sample both transition tubes. With one suboptimal order parameter $(x + y)$ FFS always samples the positive-$y$ transition tube, while with a different suboptimal order parameter $(x)$, FFS always samples the negative-$y$ transition tube. cFFS consistently samples both transition tubes.

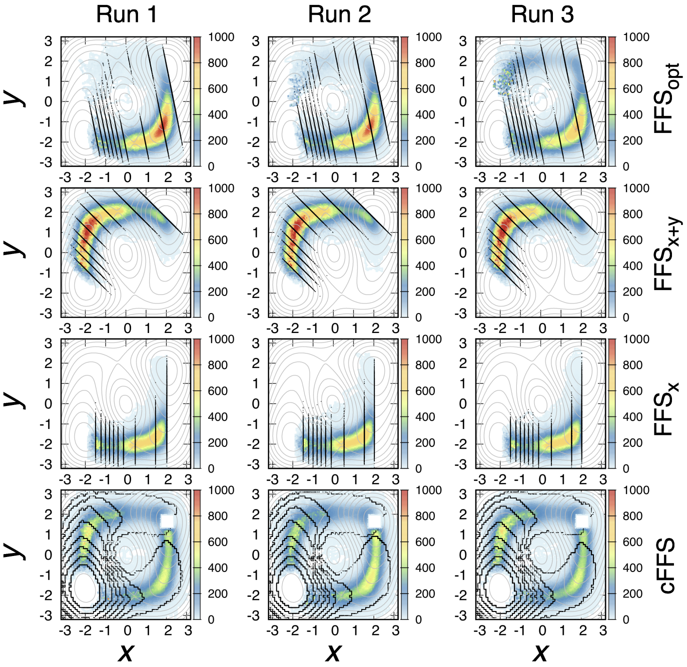

Figure B.2: Comparison of interface placement and TPE sampling generated with FFS$_{\mathrm{opt}}$, FFS$_{\mathrm{x+y}}$, FFS$_{\mathrm{x}}$, and cFFS on PES-3 at $\beta = 5.0$. Results are shown for all three independent FFS runs for each method. PES contours are shown as gray lines. Configurations at each interface are shown with black points. TPE sampling is represented by the heat map.

## B.5 Details of alanine dipeptide simulations

Alanine dipeptide was simulated in vacuum with Langevin dynamics at 300 K with the leap-frog stochastic dynamics integrator implemented in GROMACS 2018 [326]. The integration time step was 0.002 ps and $\gamma = 100$ ps$^{-1}$. Linear and angular center of mass motion was removed every step. Alanine dipeptide was represented with the AMBER99SB force field [327]. Bonds between heavy atoms and a hydrogen were constrained with LINCS [328, 329].

cFFS was tested by investigating the C$_{7\mathrm{ax}}$-to-C$_{7\mathrm{eq}}$ conformational transition, which requires surmounting an $\sim$10 $k_BT$ barrier [259]. The SLD rate constant was estimated from 25 independent simulations. Each simulation was initiated from the C$_{7\mathrm{ax}}$ basin located near $\phi = 60°$ and $\psi = -30°$. Following an energy minimization, the systems were equilibrated for 1 ns prior to the start of the production runs. Each production run was continued until the system committed to the C$_{7\mathrm{eq}}$ basin or for a maximum of 500 ns. 23 of the 25 simulations underwent the conformational transition within 500 ns. The rate constant was estimated as $k_{AB} = n_{AB}/t_A$ where $n_{AB}$ is the number of C$_{7\mathrm{ax}}$-to-C$_{7\mathrm{eq}}$ transitions and $t_A$ is the total simulation time spent in the C$_{7\mathrm{ax}}$ basin, and thus $k_{AB}^{\mathrm{SLD}} = 4.8 \times 10^6$ s$^{-1}$.

cFFS was performed for the same system. Simulation in basin $A$ was initiated from an energy minimized configuration in the C$_{7\mathrm{ax}}$ basin. The system was equilibrated for 1 ns prior to the start of a 10 ns production simulation. $\phi$ and $\psi$ were selected as the CVs for cFFS. The grid extended from $-180°$ to $180°$ in both $\phi$ and $\psi$ with periodic boundaries. A grid size of 2° was used in both $\phi$ and $\psi$. The bounds of basin $B$ were defined by examining the free energy landscape reported in the Supporting Information of Ref. 259. The bounds of basin $A$ were identified with a threshold probability density of $4.0 \times 10^{-4}$. This results in the system spending $\sim$60% of the time within the bounds of $A$ during the basin simulation. The flux from $A$ to $\lambda_0$ was calculated to be $6.15 \times 10^{10}$ s$^{-1}$. Atomic velocities were *not* regenerated at the shooting points as the stochastic dynamics allowed individual trajectories to diverge. Complete details of the cFFS run are reported in Table B.2. The total probability of reaching $B$ from $\lambda_0$ is $\sum_{i=0}^{4} P(\lambda_B|\lambda_i)P(\lambda_i|\lambda_0) = 8.15 \times 10^{-5}$ and thus $k_{AB}^{\mathrm{cFFS}} = 5.0 \times 10^6$ s$^{-1}$.

Table B.2: Alanine dipeptide cFFS details

| $i$ | $N_{\text{conf}}$ | $N_{\text{basin}}$ | $N_{\text{cross}}$ | $N_{\text{succ}}$ | $N_{\text{total}}$ | $P(\lambda_{i+1}\|\lambda_i)$ | $P(\lambda_i\|\lambda_0)$ | $P(\lambda_B\|\lambda_i)$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 615 | 9347 | 653 | 0 | 10000 | 0.0653 | 1.0 | 0.0 |
| 1 | 653 | 9394 | 606 | 0 | 10000 | 0.0606 | $6.53 \times 10^{-2}$ | 0.0 |
| 2 | 606 | 9397 | 586 | 17 | 10000 | 0.0586 | $3.96 \times 10^{-3}$ | 0.0017 |
| 3 | 582 | 6471 | 466 | 3063 | 10000 | 0.0466 | $2.32 \times 10^{-4}$ | 0.3063 |
| 4 | 466 | 3258 | 0 | 1742 | 5000 | 0.0 | $1.08 \times 10^{-5}$ | 0.3484 |

# Appendix C Supporting Information for Chapter 6

## C.1 Simulations for crystal structure identification

Simulations of pure phases were performed to generate training data for the PointNet. The pure phases were simulated in a range of temperature and pressure conditions to expose the network to conditions with varied density and magnitude of thermal fluctuations. Though temperature and pressure conditions sometimes exceeded the thermodynamic stability of the simulated phases, we confirmed that all phases remained mechanically stable for the duration of the simulations. Simulation details specific to the different systems are provided in the following sections.

### C.1.1 Lennard–Jonesium

Simulations of bulk liquid, face-centered cubic (fcc), hexagonal close-packed (hcp), and body-centered cubic (bcc) phases were performed in a range of conditions both above and below the melting point. Initial configurations for the solid phases were generated by replicating the unit cell and resulted in 16384, 14976, and 17496 atoms for the fcc, hcp, and bcc phases, respectively. The initial configuration for the liquid phase consisted of 16384 atoms randomly placed in a cubic simulation box of length $25\sigma$, where $\sigma$ is the size parameter in the LJ potential. All values for the LJ system are reported in reduced units.

Simulations of the liquid, fcc, and hcp phases were performed in the $NpT$ ensemble at a range of temperatures between 0.5 and $1.7\epsilon/k_{\mathrm{B}}$ and pressures between 0 and $15\epsilon/\sigma^3$. Simulations of the bcc phase were performed in the $NVT$ ensemble with a range of temperatures between 0.6 and 1.6 $\epsilon/k_{\mathrm{B}}$ and densities between 0.95 and $1.2\sigma^{-3}$. Each $NpT$ system was first equilibrated for $500\tau$ to the target conditions, followed by a $2000\tau$ simulation with the Bussi thermostat [330] and Parrinello-Rahman barostat [219], each with coupling constant $0.5\tau$. Since the bcc phase is unstable with respect to transformation to the close-packed phases, a slab of frozen particles in bcc arrangement was used to stabilize the crystal. Analysis was only performed on particles several layers from the frozen slab. Simulations were performed in GROMACS 2018 [326]. A time step of $0.001\tau$ was used. Group cutoff scheme was used with neighbor list updates every 10 steps and cutoff radius of $4.0\sigma$. The LJ potential was force-switched from a distance of $3.0\sigma$ to the cutoff at $3.5\sigma$.

### C.1.2 Water

All-atom simulations of water were performed for the liquid phase, five ice phases, and two guest-free hydrate phases. Liquid and ice phases were simulated at temperatures spaced between 200 K and 300 K and at pressures between 1 bar and 12000 bar. Hydrate phases were sampled at temperatures 230–270 K, with pressures -2000–1000 bar. Initial configurations for the solid phases were generated by replicating the unit cell, and resulted in the following numbers of water molecules in each system: ice Ih: 768, ice Ic: 512, ice III: 768, ice V: 1792, ice VI: 640, hydrate sI: 1242, hydrate sII: 1088. The initial liquid configuration consisted of 909 water molecules. Following an energy minimization step, systems were simulated for 25 ns in the $NpT$ ensemble at the target temperature and pressure. Temperature was maintained with the thermostat of Bussi *et al.* [330] with a coupling constant of 0.5 ps. Anisotropic (isotropic) pressure coupling was applied for the solid(liquid) phase(s) with the Berendsen barostat [217] with a coupling time constant of 5 ps. The first 5 ns of the simulation was treated as equilibration and not used for data collection.

Water was described by the TIP4P/Ice [331] model. Simulations were performed in GROMACS 2018 [326]. Dynamics were propagated by the leap-frog integrator with an integration time step of 2 fs. Linear center-of-mass motion was removed every 10 integration steps. Cutoffs for LJ and Coloumbic interactions were set to 1.0 nm. The Verlet cutoff scheme was employed with the Verlet buffer tolerance set to 0.005 [332]. Long-range electrostatics were treated with particle mesh Ewald [333]. Geometry of water molecules was maintained with SETTLE [334].

### C.1.3 Mesophases

Simulations of six mesophases were performed: liquid, lamellar, lxs, hexagonal, gyroid, and body-centered cubic. The systems were described by the model presented in Ref. 134, which is comprised of pairwise interactions using the two-body term of the Stillinger–Weber potential [185]. The systems comprise of two particle types, denoted A and B. Different mesophases form from tuning the A–B interactions and the fraction of type A, $\chi_{\mathrm{A}}$. All simulations were performed with $\varepsilon_{\mathrm{AA}} = \varepsilon_{\mathrm{BB}} = 1.0$ kcal mol$^{-1}$, $\sigma_{\mathrm{AA}} = \sigma_{\mathrm{BB}} = 1.0$ and $\sigma_{\mathrm{AB}} = 1.15$. Values other than temperature and energy are reported as dimensionless quantities. All simulations are performed at $T = 300$ K and $p = 0$. Simulations are performed in the $\chi_{\mathrm{A}} > 0.5$ portion of the phase diagram so type B is the minor component.

Guided by the phase diagram presented in Fig. 6 of Ref. 134, we select the following conditions for each phase. Liquid: $\chi_{\mathrm{A}} = 0.5$, $\varepsilon_{\mathrm{AB}} = 0.85$ kcal mol$^{-1}$, lamellar: $\chi_{\mathrm{A}} = 0.5$, $\varepsilon_{\mathrm{AB}} = 1.4$ kcal mol$^{-1}$, shifted layered crystal (lxs): $\chi_{\mathrm{A}} = 0.5$, $\varepsilon_{\mathrm{AB}} = 1.9$ kcal mol$^{-1}$, hexagonal: $\chi_{\mathrm{A}} = 0.77$, $\varepsilon_{\mathrm{AB}} = 1.8$ kcal mol$^{-1}$, gyroid: $\chi_{\mathrm{A}} = 0.67$, $\varepsilon_{\mathrm{AB}} = 1.8$ kcal mol$^{-1}$, body-centered cubic (bcc): $\chi_{\mathrm{A}} = 0.86$, $\varepsilon_{\mathrm{AB}} = 3.8$ kcal mol$^{-1}$. Except for the body-centered cubic phase, all phases were generated through nucleation from the isotropic liquid. All systems except bcc contained 16384 atoms. The bcc systems contained 14000 atoms.

Simulations were performed in GROMACS 2018 [326] using tabulated potentials. The cutoff was set to the theoretical maximum for the Stillinger–Weber potential. Equations of motion were integrated with the leap-frog integrator with a time step of 0.005. Systems were equilibrated for 500,000 steps in the $NpT$ ensemble with temperature and pressure coupling maintained by the Bussi thermostat [330] ($\tau_T = 2.0$) and Berendsen barostat [217] ($\tau_p = 4.1$), respectively. For the production simulations, temperature and pressure were maintained with the Bussi thermostat [330] and Parrinello–Rahman barostat [219] with damping constants of $\tau_T = 2.0$ and $\tau_p = 10.2$, respectively. Production simulations were performed for $2.5 \times 10^8$ steps. Only the portion of the simulations after the crystal phase had grown to occupy the entire simulation box were used for analysis.

## C.2 Simulations for hydrophobicity identification

### C.2.1 Self-assembled monolayer systems

Self-assembled monolayer (SAM) surfaces are flexible organic surfaces composed of alkane chains attached to a metal surface. All SAM surfaces were constructed to be approximately $6 \times 7$ nm with 192 alkane chains total. Each chain contains a sulfur atom attached to one end of a 10-carbon alkane chain and a terminal group at the other end, in this case $CH_3$ and OH. Sulfur atoms were restrained to positions corresponding to their hypothetical spacing when adsorbed to a Au (111) surface. The in-plane structure of the sulfur atoms was $\sqrt{3} \times \sqrt{3}$ R30 ith a 0.497 nm distance between neighboring sulfur atoms [170]. The surfaces are periodic in $x$ and $y$ directions. Partial charges were taken from the OPLS-AA force field [186]. All other bonded and nonbonded parameters were taken from the General Amber force field [335]. The surface with vacuum space on either side in the $z$ direction was equilibrated in the $NVT$ ensemble for 5 ns at 300 K. A slab of 6000

TIP3P water molecules was placed in contact with the surface terminal groups. The vacuum space above the water acts as a natural barostat, maintaining the pressure at 0 bar. The surface–water system was equilibrated in the $NVT$ ensemble (300 K) for 5 ns. Training and testing samples were collected from a subsequent production run of 25 ns in the $NVT$ ensemble (300 K). Simulations were performed in GROMACS [326] with a time step of 0.002 ps. The Bussi thermostat [330] maintained temperature with time constant $\tau_T = 0.5$ ps. Hydrogen bonds were constrained with LINCS [328]. LJ and Coulombic cutoffs were set to 1.0 nm. Particle mesh Ewald was used to calculate long-range electrostatics [336].

### C.2.2 Protein systems

Structures of hydrophobin II (PDB: 2B97) and CheY (PDB: 3CHY) were taken from the Protein Data Bank (PDB). Hydrophobin and CheY were solvated with TIP3P water in $5 \times 5 \times 5$ nm$^3$ and $8 \times 8 \times 8$ nm$^3$ simulation boxes, respectively. Four sodium counter ions were added to the CheY system. The proteins were described by the AMBER99SB-ILDN force field [337]. Heavy atoms of the proteins were position restrained and the systems were energy minimized. Following the energy minimization, the systems were equilibrated for 5 ns in the $NpT$ ensemble (300 K, 1 bar) with the protein heavy atoms position restrained. Temperature coupling was only applied to the solvent (Bussi thermostat [330], $\tau_T = 0.5$ ps). The Berendsen barostat [217] was used during equilibration with $\tau_p = 1.0$ ps. Systems were simulated in production for 25 ns in the $NpT$ ensemble (300 K, 1 bar) with no position restraints. Temperature coupling was applied with the Bussi thermostat [330] ($\tau_T = 1.0$ ps) and the Parrinello-Rahman barostat [219] ($\tau_p = 5.0$ ps). Temperature coupling was only applied to the solvent. All simulations were performed in GROMACS 2018 [326]. Equations of motion were integrated with the leap frog algorithm with a time step of 0.002 ps. LJ and Coulombic cutoffs were set to 1.0 nm. Particle mesh Ewald was used to calculate long-range electrostatics [336]. Hydrogen bonds were constrained with the LINCS algorithm [328].

# Bibliography

[1] R. J. Allen, D. Frenkel, and P. R. ten Wolde. Simulating rare events in equilibrium or nonequilibrium stochastic systems. *J. Chem. Phys.*, 124:024102, 2006.

[2] J Kyte and R F Doolittle. A simple method for displaying the hydropathic character of a protein. *J. Mol. Bio.*, 157(1):105–132, 1982.

[3] L H Kapcha and P J Rossky. A simple atomic-level hydrophobicity scale reveals protein interfacial structure. *J. Mol. Bio.*, 426(2):484–498, 2014.

[4] S Shin and A P Willard. Characterizing hydration properties based on the orientational structure of interfacial water molecules. *J. Chem. Theory Comput.*, 14(2):461–465, 2018.

[5] Juan Manuel Garca-Ruiz and Fermn Otlora. Crystal growth in geology: Patternson the rocks. In Peter Rudolph, editor, *Handbook of Crystal Growth*, Handbook of Crystal Growth, pages 1 – 43. Elsevier, Boston, second edition edition, 2015.

[6] Peter L Davies. Ice-binding proteins: a remarkable diversity of structures for stopping and starting ice growth. *Trends Biochem. Sci.*, 39(11):548–555, 2014.

[7] M Bar Dolev, I Braslavsky, and P L Davies. Ice-binding proteins and their function. *Ann. Rev. Biochem.*, 85:515–542, 2016.

[8] BJ Murray, D O'Sullivan, JD Atkinson, and ME Webb. Ice nucleation by particles immersed in supercooled cloud droplets. *Chem. Soc. Rev.*, 41(19):6519–6554, 2012.

[9] J. D. Atkinson, B. J. Murray, M. T. Woodhouse, T. F. Whale, K. J. Baustian, K. S. Carslaw, S. Dobbie, D. OSullivan, and T. L. Malkin. The importance of feldspar for ice nucleation by mineral dust in mixed-phase clouds. *Nature*, 498(7454):355, 2013.

[10] Wolfgang Beckmann. *Crystallization: basic concepts and industrial applications.* John Wiley & Sons, 2013.

[11] M J Kreder, J Alvarenga, P Kim, and J Aizenberg. Design of anti-icing surfaces: smooth, textured or slippery? *Nature Rev. Mater.*, 1(1):15003, 2016.

[12] S Zhang, J Huang, Y Cheng, H Yang, Z Chen, and Y Lai. Bioinspired surfaces with superwettability for anti-icing and ice-phobic application: Concept, mechanism, and design. *Small*, 13(48):1701867, 2017.

[13] J. Carroll. *Natural Gas Hydrates: A Guide for Engineers.* Gulf Professional Publishing, Waltham, MA, 3 edition, 2014.

[14] Adam Ballard, Jefferson Creek, Michael Eaton, Carolyn Koh, Jason Lachance, Norm McMullen, Thierry Palermo, George Shoup, Dendy Sloan, Amadeu K. Sum, and Larry Talley. Elsevier, 2011.

[15] H R Pruppacher. A new look at homogeneous ice nucleation in supercooled water drops. *J. Atmos. Sci.*, 52(11):1924–1933, 1995.

[16] P G Debenedetti. *Metastable liquids: concepts and principles.* Princeton University Press, 1996.

[17] G. C. Sosso, J. Chen, S. J. Cox, M. Fitzner, P. Pedevilla, A. Zen, and A. Michaelides. Crystal nucleation in liquids: Open questions and future challenges in molecular dynamics simulations. *Chem. Rev.*, 116:7078–7116, 2016.

[18] M. Volmer and A. Weber. Germ formation in oversaturated figures. *Z. Phys. Chem.*, 119:277–301, 1926.

[19] L. Farkas. The speed of germinitive formation in over saturated vapours. *Z. Phys. Chem.*, 125:236–242, 1927.

[20] R. Becker and W. Döring. Kinetic treatment of germ formation in supersaturated vapour. *Annalen Der Physik*, 24:719–752, 1935.

[21] J. B. Zeldovich. On the theory of new phase formation, cavitation. *Acta Physicochim. URSS*, 18:1–22, 1943.

[22] J Frenkel. *Kinetic Theory of Liquids*, chapter 7. Dover, New York, 1955.

[23] M. Volmer and A. Weber. Particle formation and particle action as a special case of heterogeneous catalysis. *Zeits. f. Elektrochemie*, 35:555–561, 1929.

[24] D Turnbull and B Vonnegut. Nucleation catalysis. *Industrial & Engineering Chemistry*, 44(6):1292–1298, 1952.

[25] B Peters. Common features of extraordinary rate theories. *J. Phys. Chem. B*, 119(21):6349–6356, 2015.

[26] D Gebauer, M Kellermeier, J D Gale, L Bergström, and H Cölfen. Pre-nucleation clusters as solute precursors in crystallisation. *Chemical Society Reviews*, 43(7):2348–2371, 2014.

[27] L. C. Jacobson, W. Hujo, and V. Molinero. Amorphous precursors in the nucleation of clathrate hydrates. *J. Am. Chem. Soc.*, 132:11806–11811, 2010.

[28] D Gebauer, A Völkel, and H Cölfen. Stable prenucleation calcium carbonate clusters. *Science*, 322(5909):1819–1822, 2008.

[29] P G Vekilov. The two-step mechanism of nucleation of crystals in solution. *Nanoscale*, 2(11):2346–2357, 2010.

[30] J. T. Berryman, M. Anwar, S. Dorosz, and T. Schilling. The early crystal nucleation process in hard spheres shows synchronised ordering and densification. *J. Chem. Phys.*, 145(21):211901, 2016.

[31] J. Russo and H. Tanaka. Crystal nucleation as the ordering of multiple order parameters. *J. Chem. Phys.*, 145:211801, 2016.

[32] D Frenkel and B Smit. *Understanding molecular simulation: from algorithms to applications.* Elsevier, 2001.

[33] M P Allen and D J Tildesley. *Computer simulation of liquids.* Oxford University Press, 2017.

[34] David Chandler. *Introduction to modern statistical mechanics.* Oxford University Press, 1987.

[35] B J Alder and T E Wainwright. Phase transition for a hard sphere system. *J. Chem. Phys.*, 27(5):1208–1209, 1957.

[36] U Gasser, E R Weeks, A Schofield, PN Pusey, and DA Weitz. Real-space imaging of nucleation and growth in colloidal crystallization. *Science*, 292(5515):258–262, 2001.

[37] V J Anderson and H NW Lekkerkerker. Insights into phase transition kinetics from colloid science. *Nature*, 416(6883):811, 2002.

[38] U Gasser. Crystallization in three-and two-dimensional colloidal suspensions. *J. Phys. Cond. Mat.*, 21(20):203101, 2009.

[39] S Auer and D Frenkel. Prediction of absolute crystal-nucleation rate in hard-sphere colloids. *Nature*, 409(6823):1020, 2001.

[40] L Filion, M Hermes, R Ni, and M Dijkstra. Crystal nucleation of hard spheres using molecular dynamics, umbrella sampling, and forward flux sampling: A comparison of simulation techniques. *J. Chem. Phys.*, 133(24):244115, 2010.

[41] T Kawasaki and H Tanaka. Formation of a crystal nucleus from liquid. *Proc. Natl. Acad. Sci.*, 107(32):14036–14041, 2010.

[42] L. Filion, R. Ni, D. Frenkel, and M. Dijkstra. Simulation of nucleation in almost hard-sphere colloids: The discrepancy between experiment and simulation persists. *J. Chem. Phys.*, 134(13):134901, 2011.

[43] P R Ten Wolde, M J Ruiz-Montero, and D Frenkel. Numerical evidence for bcc ordering at the surface of a critical fcc nucleus. *Phys. Rev. Lett.*, 75(14):2714, 1995.

[44] D Moroni, P R Ten Wolde, and P G Bolhuis. Interplay between structure and size in a critical crystal nucleus. *Phys. Rev. Lett.*, 94(23):235703, 2005.

[45] C Desgranges and J Delhommelle. Controlling polymorphism during the crystallization of an atomic fluid. *Phys. Rev. Lett.*, 98(23):235502, 2007.

[46] D Richard and T Speck. Crystallization of hard spheres revisited. i. extracting kinetics and free energy landscape from forward flux sampling. *J. Chem. Phys.*, 148(12):124110, 2018.

[47] D Richard and T Speck. Crystallization of hard spheres revisited. ii. thermodynamic modeling, nucleation work, and the surface of tension. *J. Chem. Phys.*, 148(22):224102, 2018.

[48] M. Dalvi-Isfahan, N. Hamdami, E. Xanthakis, and A. Le-Bail. Review on the control of ice nucleation by ultrasound waves, electric and magnetic fields. *Journal of Food Engineering*, 195:222–234, 2017.

[49] M Matsumoto, S Saito, and I Ohmine. Molecular dynamics simulation of the ice nucleation and growth process leading to water freezing. *Nature*, 416(6879):409, 2002.

[50] T. Li, D. Donadio, G. Russo, and G. Galli. Homogeneous ice nucleation from supercooled water. *Phys. Chem. Chem. Phys.*, 13(44):19807–19813, 2011.

[51] A. Haji-Akbari, R. S. DeFever, S. Sarupria, and P. G. Debenedetti. Suppression of sub-surface freezing in free-standing thin films of a coarse-grained model of water. *Phys. Chem. Chem. Phys.*, 16(47):25916–25927, 2014.

[52] J. R. Espinosa, E. Sanz, C. Valeriani, and C. Vega. Homogeneous ice nucleation evaluated for several water models. *J. Chem. Phys.*, 141(18):18C529, 2014.

[53] A. Haji-Akbari and P. G. Debenedetti. Direct calculation of ice homogeneous nucleation rate for a molecular model of water. *Proc. Natl. Acad. Sci. USA*, 112(34):10582–10588, 2015.

[54] L. Lupi, A. Hudait, B. Peters, M. Grünwald, R. G. Mullen, A. H. Nguyen, and V. Molinero. Role of stacking disorder in ice nucleation. *Nature*, 551(7679):218, 2017.

[55] B. Glatz and S. Sarupria. The surface charge distribution affects the ice nucleating efficiency of silver iodide. *J. Chem. Phys.*, 145(21):211924, 2016.

[56] S. A. Zielke, A. K. Bertram, and G. N. Patey. Simulations of ice nucleation by model agi disks and plates. *J. Phys. Chem. B*, 120(9):2291–2299, 2016.

[57] G. C. Sosso, G. A. Tribello, A. Zen, P. Pedevilla, and A. Michaelides. Ice formation on kaolinite: Insights from molecular dynamics simulations. *J. Chem. Phys.*, 145(21):211927, 2016.

[58] G. C. Sosso, T. Li, D. Donadio, G. A. Tribello, and A. Michaelides. Microscopic mechanism and kinetics of ice formation at complex interfaces: Zooming in on kaolinite. *J. Phys. Chem. Lett.*, 7(13):2350–2355, 2016.

[59] S. A. Zielke, A. K. Bertram, and G. N. Patey. Simulations of ice nucleation by kaolinite (001) with rigid and flexible surfaces. *J. Phys. Chem. B*, 120(8):1726–1734, 2016.

[60] B. Glatz and S. Sarupria. Heterogeneous ice nucleation: Interplay of surface properties and their impact on water orientations. *Langmuir*, 34(3):1190–1198, 2017.

[61] G. C. Sosso, T. F. Whale, M. A. Holden, P. Pedevilla, B. J. Murray, and A. Michaelides. Unravelling the origins of ice nucleation on organic crystals. *Chem. Sci.*, 9(42):8077–8088, 2018.

[62] B. C. Knott, V. Molinero, M. F. Doherty, and B. Peters. Homongeneous nucleation of methane hydrates: Unrealistic under realistic conditions. *J. Am. Chem. Soc.*, 134:19544–19547, 2012.

[63] J. R. Espinosa, C. Vega, C. Valeriani, and E. Sanz. Seeding approach to crystal nucleation. *J. Chem. Phys.*, 144(3):034501, 2016.

[64] N. E. R. Zimmermann, B. Vorselaars, J. R. Espinosa, D. Quigley, W. R. Smith, E. Sanz, C. Vega, and B. Peters. NaCl nucleation from brine in seeded simulations: Sources of uncertainty in rate estimates. *J. Chem. Phys.*, 148(22):222838, 2018.

[65] P. Pedevilla, M. Fitzner, G. C. Sosso, and A. Michaelides. Heterogeneous seeded molecular dynamics as a tool to probe the ice nucleating ability of crystalline surfaces. *J. Chem. Phys.*, 149(7):072327, 2018.

[66] T. Yuan, R. S. DeFever, and S. Sarupria. Rseeds: Rigid seeding method for studying heterogeneous crystal nucleation. *under review*, 2019.

[67] Y. F. Makogon. *Hydrates of Hydrocarbons*. Pennwell Books, Tulsa, OK, 1 edition, 1997.

[68] E. D. Sloan and C. A. Koh. *Clathrate Hydrates of Natural Gases*. CRC Press/Taylor-Francis, Boca Raton, FL, 3 edition, Jan 2008.

[69] Peter V Hobbs. *Ice physics*. Oxford university press, 2010.

[70] H. Davy. Gas hydrates. *Phil. Trans. Roy. Soc. Lond.*, 101(1):71–81, 1811.

[71] E. D. Sloan. Fundamental principles and applications of natural gas hydrates. *Nature*, 426:353–363, 2003.

[72] M A Kelland. History of the development of low dosage hydrate inhibitors. *Energy & fuels*, 20(3):825–847, 2006.

[73] A Perrin, O M Musa, and J W Steed. The chemistry of low dosage clathrate hydrate inhibitors. *Chem. Soc. Rev.*, 42(5):1996–2015, 2013.

[74] A. V. Milkov. Global estimates of hydrate-bound gas in marine sediments: How much is really out there? *Earth-Science Rev.*, 66:183–197, 2004.

[75] R. C. Chong, S. H. B. Yang, P. Babu, P. Linga, and L. Xiao-Sen. Review of natural gas hydrates as an energy resource: Prospects and challenges. *Appl. Energ.*, 162:1633–1652, 2016.

[76] Y. F. Makogon. Natural gas hydrates – a promising source of energy. *J. Nat. Gas Sci. Ener.*, 2:49–59, 2010.

[77] C. A. Koh, A. K. Sum, and E. D. Sloan. Gas hydrates: Unlocking the energy from icy cages. *J. Appl. Phys.*, 106:9, 2009.

[78] Y. F. Makogon, S. A. Holditch, and T. Y. Makogon. Natural gas-hydrates – a potential energy source for the 21st century. *J. Petrol. Sci. Eng.*, 56:14–31, 2007.

[79] Timothy Collett, Jang-Jun Bahk, Matt Frye, Dave Goldberg, Jarle Husebo, Carolyn Koh, Mitch Malone, Craig Shipp, and Marta Torres. Historical methane hydrate project review. Technical report, Consortium for Ocean Leadership, 2013.

[80] K Ohgaki. A proposal for gas storage on the bottom of the ocean using gas hydrates. *Int. Chem. Eng.*, 34:417–419, 1994.

[81] X. Lang, S. Fan, and Y. Wang. Intensification of methane and hydrogen storage in clathrate hydrate and future prospect. *J. Natural Gas. Chem.*, 19:203–209, 2010.

[82] H. P. Veluswamy, A. J. H. Wong, P. Babu, R. Kumar, S. Kulprathipanja, P. Rangsunvigit, and P. Linga. Rapid methane hydrate formation to develop a cost effective large scale energy storage system. *Chem. Eng. J.*, 290:161–173, 2016.

[83] L J Florusse, C J Peters, J Schoonman, K C Hester, C A Koh, S F Dec, K N Marsh, and E D Sloan. Stable low-pressure hydrogen clusters stored in a binary clathrate hydrate. *Science*, 306(5695):469–471, 2004.

[84] H. Lee, J. Lee, D. Y. Kim, J. Park, Y. Seo, H. Zeng, I. L. Moudrakovski, C. I. Ratcliffe, and J. A. Ripmeester. Tuning clathrate hydrates for hydrogen storage. *Nature*, 434:743–746, 2005.

[85] A. Eslamimanesh, A. H. Mohammadi, D. Richon, P. Naidoo, and D. Ramjugernath. Application of gas hydrate formation in separation processes: a review of experimental studies. *J. Chem. Thermodyn.*, 46:62–71, 2012.

[86] S. D. Kenarsari, D. Yang, G. Jiang, S. Zhang, J. Wang, A. G. Russell, Q. Wei, and M. Fan. Review of recent advances in carbon dioxide separation and capture. *RSC Adv.*, 3:22739–22773, 2013.

[87] P. Babu, H. W. N. Ong, and P. Linga. A systematic kinetic study to evaluate the effect of tetrahydrofuran on the clathrate process for pre-combustion capture of carbon dioxide. *Energy*, 94:431–442, 2016.

[88] X. M. Peng, Y. F. Hu, Y. S. Liu, C. W. Jin, and H. J. Lin. Separation of ionic liquids from dilute aqueous solutions using the method based on $co_2$ hydrates. *J. Natural Gas Chem.*, 19(1):81–85, 2010.

[89] H Mimachi, M Takahashi, S Takeya, Y Gotoh, A Yoneyama, K Hyodo, T Takeda, and T Murayama. Effect of long-term storage and thermal history on the gas content of natural gas hydrate pellets under ambient pressure. *Energy & Fuels*, 29(8):4827–4834, 2015.

[90] R L Christiansen and E D Sloan. Mechanisms and kinetics of hydrate formation. *Ann. NY Acad. Sci.*, 715(1):283–305, 1994.

[91] R. Radhakrishnan and B. L. Trout. A new approach for studying nucleation phenomena using molecular simulations: Application to $co_2$ hydrate clathrates. *J. Chem. Phys.*, 117(4):1786–1796, 2002.

[92] G. Guo, M. Li, Y. Zhang, and C. Wu. Why can water cages adsorb aqueous methane? a potential of mean force calculation on hydrate nucleation mechanisms. *Phys. Chem. Chem. Phys.*, 11(44):10427–10437, 2009.

[93] R. W. Hawtin, D. Quigley, and P. M. Rodger. Gas hydrate nucleation and cage formation at a water/methane interface. *Phys. Chem. Chem. Phys.*, 10(32):4853–4864, 2008.

[94] M. R. Walsh, C. A. Koh, E. D. Sloan, A. K. Sum, and D. T. Wu. Microsecond simulations of spontaneous methane hydrate nucleation and growth. *Science*, 326:1095–1098, 2009.

[95] L. C. Jacobson, W. Hujo, and V. Molinero. Nucleation pathways of clathrate hydrates: Effect of guest size and solubility. *J. Phys. Chem. B*, 114:13796–13807, 2010.

[96] J. Vatamanu and P. G. Kusalik. Observation of two-step nucleation in methane hydrates. *Phys. Chem. Chem. Phys.*, 12:15065–15072, 2010.

[97] T. Koga, J. Wong, M. K. Endoh, D. Mahajan, C. Gutt, and S. K. Satija. Hydrate formation at the methane/water interface on the molecular scale. *Langmuir*, 26:4627–4630, 2010.

[98] M. R. Walsh, G. T. Beckham, C. A. Koh, E. D. Sloan, D. T. Wu, and A. K. Sum. Methane hydrate nucleation rates from molecular dynamics simulations: effects of aqueous methane concentration, interfacial curvature, and system size. *J. Phys. Chem. C*, 115:21241–21248, 2011.

[99] S. Liang and P. G. Kusalik. Exploring nucleation of $H_2S$ hydrates. *Chem. Sci.*, 2:1286–1292, 2011.

[100] M. R. Walsh, G. T. Beckham, C. A. Koh, D. E. Sloan, D. T. Wu, and A. K. Sum. Methane hydrate nucleation rates from molecular dynamics simulations: effects of aqueous methane concentration, interfacial curvature and system size. *J. Phys. Chem. C*, 115:21241–21248, 2011.

[101] S. Sarupria and P. G. Debenedetti. Homogeneous nucleation of methane hydrate in microsecond molecular dynamics simulations. *J. Phys. Chem. Lett.*, 3:2942–2947, 2012.

[102] P. Pirzadeh and P. G. Kusalik. Molecular insights into clathrate hydrate nucleation at an ice–solution interface. *J. Am. Chem. Soc.*, 135:7278–7287, 2013.

[103] S. Liang and P. G. Kusalik. Nucleation of gas hydrates within constant energy systems. *J. Phys. Chem. B*, 117:1403–1410, 2013.

[104] Y. Bi and T. Li. Probing methane hydrate nucleation through the forward flux sampling method. *J. Phys. Chem. B*, 118:13324–13332, 2014.

[105] M. Lauricella, S. Meloni, N. J. English, B. Peters, and G. Ciccotti. Methane clathrate hydrate nucleation mechanism by advanced molecular simulations. *J. Phys. Chem. C*, 118:22847–22857, 2014.

[106] Z. Zhang, M. R. Walsh, and G. Guo. Microcanonical molecular simulations of methane hydrate nucleation and growth: evidence that direct nucleation to sI hydrate is among the multiple nucleation pathways. *Phys. Chem. Chem. Phys.*, 17:8870–8876, 2015.

[107] E. Małolepsza and T. Keyes. Pathways through equilibrated states with coexisting phases for gas hydrate formation. *J. Phys. Chem. B*, 119(52):15857–15865, 2015.

[108] Y. Bi, A. Porras, and T. Li. Free energy landscape and molecular pathways of gas hydrate nucleation. *J. Chem. Phys.*, 145:211909, 2016.

[109] Z. Zhang, C. Liu, M. R. Walsh, and G. Guo. Effects of ensembles of methane hydrate nucleation kinetics. *Phys. Chem. Chem. Phys.*, 18:15602–15608, 2016.

[110] Z. He, K. M. Gupta, P. Linga, and J. Jiang. Molecular insights into the nucleation and growth of ch4 and co2 mixed hydrates from microsecond simulations. *J. Phys. Chem. C*, 120(44):25225–25236, 2016.

[111] J. Wu, L. Chen, Y. Chen, and S. Lin. Molecular dynamics study on the nucleation of methane + tetrahydrofuran mixed guest hydrate. *Phys. Chem. Chem. Phys.*, 18(15):9935–9947, 2016.

[112] T. Yagasaki, M. Matsumoto, and H. Tanaka. Formation of clathrate hydrates of water-soluble guest molecules. *J. Phys. Chem. C*, 120(38):21512–21521, 2016.

[113] Z He, P Linga, and J Jiang. What are the key factors governing the nucleation of co 2 hydrate? *Phys. Chem. Chem. Phys.*, 19(24):15657–15661, 2017.

[114] D. Bai, G. Chen, X. Zhang, and W. Wang. Microsecond molecular dynamics simulations of the kinetic pathways of gas hydrate formation from solid surfaces. *Langmuir*, 27:5961–5967, 2011.

[115] D. Bai, G. Chen, X. Zhang, and W. Wang. Nucleation of the $CO_2$ hydrate from three-phase contact lines. *Langmuir*, 28:7730–7736, 2012.

[116] S. Liang and P. G. Kusalik. The nucleation of gas hydrates near silica surfaces. *Can. J. Chem.*, 93:791–798, 2014.

[117] D. Bai, G. Chen, X. Zhang, A. K. Sum, and W. Wang. How properties of solid surfaces modulate the nucleation of gas hydrate. *Sci. Rep.*, 5:12747, 2015.

[118] Z He, P Linga, and J Jiang. Ch4 hydrate formation between silica and graphite surfaces: Insights from microsecond molecular dynamics simulations. *Langmuir*, 33(43):11956–11967, 2017.

[119] J. Juraszek and P. G. Bolhuis. Sampling the multiple folding mechanisms of trp-cage in explicit solvent. *Proc. Natl. Acad. Sci. USA*, 103(43):15859–15864, 2006.

[120] C Dellago, P G Bolhuis, and D Chandler. On the calculation of reaction rate constants in the transition path ensemble. *J. Chem. Phys.*, 110(14):6617–6625, 1999.

[121] R. G. Mullen, J. Shea, and B. Peters. Transmission coefficients, committors, and solvent coordinates in ion-pair dissociation. *J. Chem. Theory Comput.*, 10(2):659–667, 2014.

[122] Daniele Moroni. *Efficient sampling of rare event pathways.* PhD thesis, 2005.

[123] T. S. van Erp. *Dynamical Rare Event Simulation Techniques for Equilibrium and Nonequilibrium Systems*, chapter 2, pages 27–60. Wiley-Blackwell, 2012.

[124] G. M. Torrie and J. P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.*, 23(2):187–199, 1977.

[125] A. Laio and M. Parrinello. Escaping free-energy minima. *Proc. Natl. Acad. Sci. USA*, 99(20):12562–12566, 2002.

[126] A Barducci, G Bussi, and M Parrinello. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Phys. Rev. Lett.*, 100(2):020603, 2008.

[127] B. Peters and B. L. Trout. Obtaining reaction coordinates by likelihood maximization. *J. Chem. Phys.*, 125:054108, 2006.

[128] B. Peters, G. T. Beckham, and B. L. Trout. Extensions to the likelihood maximization approach for finding reaction coordinates. *J. Chem. Phys.*, 127(3):034109, 2007.

[129] R. J. Allen, P. B. Warren, and P. R. ten Wolde. Sampling rare switching events in biochemical networks. *Phys. Rev. Lett.*, 94:018104, 2005.

[130] A. Reinhardt and J. P. K. Doye. Free energy landscapes for homogeneous nucleation of ice for a monatomic water model. *J. Chem. Phys.*, 136(5):054501, 2012.

[131] R. S. DeFever and S. Sarupria. Nucleation mechanism of clathrate hydrates of water-soluble guest molecules. *J. Chem. Phys.*, 147(20):204503, 2017.

[132] M. Fitzner, G. C. Sosso, F. Pietrucci, S. Pipolo, and A. Michaelides. Pre-critical fluctuations and what they disclose about heterogeneous crystal nucleation. *Nat. Commun.*, 8(1):2257, 2017.

[133] L. Lupi, B. Peters, and V. Molinero. Pre-ordering of interfacial water in the pathway of heterogeneous ice nucleation does not lead to a two-step crystallization mechanism. *J. Chem. Phys.*, 145(21):211910, 2016.

[134] L. Lupi, R. Hanscam, Y. Qiu, and V. Molinero. Reaction coordinate for ice crystallization on a soft surface. *J. Chem. Phys. Lett.*, 8(17):4201–4205, 2017.

[135] B. Peters. Reaction coordinates and mechanistic hypothesis tests. *Ann. Rev. Phys. Chem.*, 67:669–690, 2016.

[136] T. S. van Erp. Efficiency analysis of reaction rate calculation methods using analytical models i: The two-dimensional sharp barrier. *J. Chem. Phys.*, 125(17):174106, 2006.

[137] C. H. Bennett. *Molecular Dynamics and Transition State Theory: The Simulation of Infrequent Events*, chapter 4, pages 63–97.

[138] D Chandler. Statistical mechanics of isomerization dynamics in liquids and the transition state approximation. *J. Chem. Phys.*, 68(6):2959–2970, 1978.

[139] J Keck. Statistical investigation of dissociation cross-sections for diatoms. *Discuss. Faraday Soc.*, 33:173–182, 1962.

[140] H Eyring. The activated complex in chemical reactions. *J. Chem. Phys.*, 3(2):107–115, 1935.

[141] E Wigner. The transition state method. *Trans. Faraday Soc.*, 34:29–41, 1938.

[142] P. G. Bolhuis and C. Dellago. *Trajectory-Based Rare Event Simulations*, chapter 3, pages 111–210. Wiley-Blackwell, 2010.

[143] C. Dellago, P. G. Bolhuis, F. S. Csajka, and D. Chandler. Transition path sampling and the calculation of rate constants. *J. Chem. Phys.*, 108(5):1964–1977, 1998.

[144] E. E. Borrero and F. A. Escobedo. Reaction coordinates and transition pathways of rare events via forward flux sampling. *J. Chem. Phys.*, 127:164101, 2007.

[145] P G Bolhuis, D Chandler, C Dellago, and P L Geissler. Transition path sampling: Throwing ropes over rough mountain passes, in the dark. *Ann. Rev. Phys. Chem.*, 53(1):291–318, 2002.

[146] T. S. van Erp, D. Moroni, and P. G. Bolhuis. A novel path sampling method for the calculation of rate constants. *J. Chem. Phys.*, 118(17):7762–7774, 2003.

[147] A Lervik, E Riccardi, and T S van Erp. Pyretis: A well-done, medium-sized python library for rare events. *J. Comput. Chem.*, 38(28):2439–2451, 2017.

[148] D WH Swenson, J Prinz, F Noe, J D Chodera, and P G Bolhuis. Openpathsampling: A python framework for path sampling simulations. 1. basics. *J. Chem. Theory Comput.*, 2018.

[149] D WH Swenson, J Prinz, F Noe, J D Chodera, and P G Bolhuis. Openpathsampling: A python framework for path sampling simulations. 2. building and customizing path ensembles and sample schemes. *J. Chem. Theory Comput.*, 15(2):837–856, 2018.

[150] M Villen-Altamirano and J Villen-Altamirano. Restart: A method for accelerating rare event simulations. *Queueing, Performance and Control in ATM*, pages 71–76, 1991.

[151] Manuel Villén-Altamirano, A Martínez-Marrón, J Gamo, and F Fernández-Cuesta. Enhancement of the accelerated simulation method restart by considering multiple thresholds. In *Teletraffic Science and Engineering*, volume 1, pages 797–810. 1994.

[152] Manuel Villén-Altamirano and Jose Villen-Altamirano. Analysis of restart simulation: Theoretical basis and sensitivity study. *European Transactions on Telecommunications*, 13(4):373–385, 2002.

[153] R. J. Allen, C. Valeriani, and P. R. ten Wolde. Forward flux sampling for rare event simulations. *J. Phys. Condens. Mat.*, 21(46):463102, 2009.

[154] J. Juraszek and P. G. Bolhuis. Rate constant and reaction coordinate of trp-cage folding in explicit water. *Biophys. J.*, 95(9):4246–4257, 2008.

[155] C. Velez-Vega, E. E. Borrero, and F. A. Escobedo. Kinetics and reaction coordinate for the isomerization of alanine dipeptide by a forward flux sampling protocol. *J. Chem. Phys.*, 130(22):225101, 2009.

[156] C. Velez-Vega, E. E. Borrero, and F. A. Escobedo. Kinetics and mechanism of the unfolding native-to-loop transition of Trp-cage in explicit solvent via optimized forward flux sampling simulations. *J. Chem. Phys.*, 133:105103, 2010.

[157] R. Cabriolu and T. Li. Ice nucleation on carbon surface supports the classical theory for heterogeneous nucleation. *Phys. Rev. E*, 91(5):052402, 2015.

[158] T E Booth and J S Hendricks. Importance estimation in forward monte carlo calculations. *Nuclear Technology-Fusion*, 5(1):90–100, 1984.

[159] PG Melnik-Melnikov and ES Dekhtyaruk. Rare events probabilities estimation by russian roulette and splitting simulation technique. *Probabilistic engineering mechanics*, 15(2):125–129, 2000.

[160] E E Borrero and F A Escobedo. Optimizing the sampling and staging for simulations of rare events via forward flux sampling schemes. *J. Chem. Phys.*, 129(2):024115, 2008.

[161] K Kratzer, A Arnold, and R J Allen. Automatic, optimized interface placement in forward flux sampling simulations. *J. Chem. Phys.*, 138(16):164112, 2013.

[162] T. S. van Erp. Efficient path sampling on multiple reaction channels. *Comput. Phys. Commun.*, 179(1-3):34–40, 2008.

[163] R S DeFever and S Sarupria. Surface chemistry effects on heterogeneous clathrate hydrate nucleation: A molecular dynamics study. *J. Chem. Thermodyn.*, 117:205–213, 2018.

[164] Z. M. Aman and C. A. Koh. Interfacial phenomena in gas hydrate systems. *Chem. Soc. Rev.*, 45:1678–1690, 2016.

[165] A. K. Sum, C. A. Koh, and E. D. Sloan. Clathrate hydrates: From laboratory science to engineering practice. *Ind. Eng. Chem. Res.*, 48:7457–7465, 2009.

[166] P. Warrier, M. N. Khan, V. Srivastava, C. M. Maupin, and C. A. Koh. Overview: Nucleation of clathrate hydrates. *J. Chem. Phys.*, 145:211705, 2016.

[167] N. J. English and J. M. D. MacElroy. Perspectives on molecular simulation of clathrate hydrates: Progress, prospects and challenges. *Chem. Eng. Sci.*, 121:133–156, 2015.

[168] B. C. Barnes and A. K. Sum. Advances in molecular simulations of clathrate hydrates. *Curr. Opin. Chem. Eng.*, 2:184–190, 2013.

[169] D Turnbull. Kinetics of heterogeneous nucleation. *J. Chem. Phys.*, 18:198–203, 1950.

[170] J. C. Love, L. A. Estroff, J. K. Kriebel, R. G. Nuzzo, and G. M. Whitesides. Self-assembled monolayers of thiolates on metals as a form of nanotechnology. *Chem. Rev.*, 105:1103–1170, 2005.

[171] M. Maccarini, R. Steitz, M. Himmelhaus, J. Fick, S. Tatur, M. Wolff, M. Grunze, J. Janecek, and R. R. Netz. Density depletion at solid–liquid interfaces: a neutron reflectivity study. *Langmuir*, 23:598–608, 2007.

[172] D. Schwendel, T. Hayashi, R. Dahint, A. Pertsin, M. Grunze, R. Steitz, and F. Schreiber. Interaction of water with self-assembled monolayers: Neutron reflectivity measurements of the water density in the interface region. *Langmuir*, 19:2284–2293, 2003.

[173] R. Godawat, S. N. Jamadagni, and S. Garde. Characterizing hydrophobicity of interfaces by using cavity formation, solute binding, and water correlations. *Proc. Natl. Acad. Sci. U.S.A.*, 106:15119–15124, 2009.

[174] V. Molinero and E. B. Moore. Water modeled as an intermediate element between Carbon and Silicon. *J. Phys. Chem. B*, 113:4008–4016, 2009.

[175] J. Lu, Y. Qiu, R. Baron, and V. Molinero. Coarse-graining of TIP4P/2005, TIP4P-Ew, SPC/E, and TIP3P to monatomic anisotropic water models using relative entropy minimization. *J. Chem. Theory Comput.*, 10:4104–4120, 2014.

[176] J. R. Espinosa, C. Navarro, E. Sanz, C. Valeriani, and C. Vega. On the time required to freeze water. *J. Chem. Phys.*, 145(21):211922, 2016.

[177] A. K. Metya, J. K. Singh, and F. Müller-Plathe. Ice nucleation on nanotextured surfaces: the influence of surface fraction, pillar height and wetting states. *Phys. Chem. Chem. Phys.*, 18(38):26796–26806, 2016.

[178] Y. Bi, R. Cabriolu, and T. Li. Heterogeneous ice nucleation controlled by the coupling of surface crystallinity and surface hydrophilicity. *J. Phys. Chem. C*, 120(3):1507–1514, 2016.

[179] S. J. Cox, S. M. Kathmann, B. Slater, and A. Michaelides. Molecular simulations of heterogeneous ice nucleation. I. Controlling ice nucleation through surface hydrophilicity. *J. Chem. Phys.*, 142:184704, 2015.

[180] T. Li, D. Donadio, and G. Galli. Ice nucleation at the nanoscale probes no man's land of water. *Nat. Commun.*, 4:1887, 2013.

[181] Y. Qiu, N. Odendahl, A. Hudait, R. Mason, A. K. Bertram, F. Paesani, P. J. DeMott, and V. Molinero. Ice nucleation efficiency of hydroxylated organic surfaces is controlled by their structural fluctuations and mismatch to ice. *J. Am. Chem. Soc.*, 139:3052–3064, 2017.

[182] Y. Bi, B. Cao, and T. Li. Enhanced heterogeneous ice nucleation by special surface geometry. *Nat. Commun.*, 8:15372, 2017.

[183] W. L. Jorgensen, J. D. Madura, and C. J. Swenson. Optimized intermolecular potential functions for liquid hydrocarbons. *J. Am. Chem. Soc.*, 106:6638–6646, 1984.

[184] W. L. Jorgensen. Optimized intermolecular potential functions for liquid alcohols. *J. Phys. Chem.*, 90:1276–1284, 1986.

[185] F. H. Stillinger and T. A. Weber. Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B*, 31:5262–5271, 1985.

[186] W. L. Jorgensen, D. S. Maxwell, and J. Tirado-Rives. Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids. *J. Am. Chem. Soc.*, 118:11225–11236, 1996.

[187] T. Werder, J. H. Walther, R. L. Jaffe, T. Halicioglu, and P. Koumoutsakos. On the water–carbon interaction for use in molecular dynamics simulations of graphite and carbon nanotubes. *J. Phys. Chem. B*, 107:1345–1352, 2003.

[188] S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.*, 117:1–19, 1995.

[189]

[190] William Humphrey, Andrew Dalke, and Klaus Schulten. VMD – Visual Molecular Dynamics. *J. Mol. Graphics*, 14:33–38, 1996.

[191] B. C. Barnes, G. T. Beckham, D. T. Wu, and A. K. Sum. Two-component order parameter for quantifying clathrate hydrate nucleation and growth. *J. Chem. Phys.*, 140:164506, 2014.

[192] L. A. Báez and P. Clancy. Computer simulation of the crystal growth and dissolution of natural gas hydrates. *Ann. N. Y. Acad. Sci.*, 715:177–186, 1994.

[193] P. R. ten Wolde, M. J. Ruiz-Montero, and D. Frenkel. Simulation of homogeneous crystal nucleation close to coexistence. *Faraday Discuss.*, 104:93–110, 1996.

[194] L. C. Jacobson, M. Matsumoto, and V. Molinero. Order parameters for the multistep crystallization of clathrate hydrates. *J. Chem. Phys.*, 135:074501, 2011.

[195] D. Kashchiev and A. Firoozabadi. Driving force for crystallization of gas hydrates. *J. Cryst. Growth*, 241:220 – 230, 2002.

[196] F. H. Stillinger. Structure in aqueous solutions of nonpolar solutes from the standpoint of scaled-particle theory. *J. Solution Chem.*, 2:141–158, 1973.

[197] K. Lum, D. Chandler, and J. D. Weeks. Hydrophobicity at small and large length scales. *J. Phys. Chem. B*, 103:4570–4577, 1999.

[198] D. Chandler. Interfaces and the driving force of hydrophobic assembly. *Nature*, 437:640–647, 2005.

[199] S. Sarupria and S. Garde. Quantifying water density fluctuations and compressibility of hydration shells of hydrophobic solutes and proteins. *Phys. Rev. Lett.*, 103:037803, 2009.

[200] P. M. Rodger, T. R. Forester, and W. Smith. Simulations of the methane hydrate/methane gas interface near hydrate forming conditions. *Fluid Phase Equilibr.*, 116:326–332, 1996.

[201] S. Vembanur, A. J. Patel, S. Sarupria, and S. Garde. On the thermodynamics and kinetics of hydrophobic interactions at interfaces. *J. Phys. Chem. B*, 117:10261–10270, 2013.

[202] G. Fraux and J. P. K. Doye. Note: Heterogeneous ice nucleation on silver-iodide-like surfaces. *J. Chem. Phys.*, 141:216101, 2014.

[203] B. C. Barnes, B. C. Knott, G. T. Beckham, D. T. Wu, and A. K. Sum. Reaction coordinate of incipient methane clathrate hydrate nucleation. *J. Phys. Chem. B*, 118(46):13236–13243, 2014.

[204] G. Guo, Y. Zhang, C. Liu, and K. Li. Using the face-saturated incomplete cage analysis to quantify the cage compositions and cage linking structures of amorphous phase hydrates. *Phys. Chem. Chem. Phys.*, 13(25):12048–12057, 2011.

[205] K. Wm. Hall, Z. Zhang, and P. G. Kusalik. Unraveling mixed hydrate formation: Microscopic insights into early stage behavior. *J. Phys. Chem. B*, 120(51):13218–13223, 2016.

[206] A Kumar, N Daraboina, R Kumar, and P Linga. Experimental investigation to elucidate why tetrahydrofuran rapidly promotes methane hydrate formation kinetics: applicable to energy storage. *J. Phys. Chem. C*, 120(51):29062–29068, 2016.

[207] Apache Hadoop. `http://hadoop.apache.org`.

[208] T. White. *Hadoop: The definitive guide.* O'Reilly Media, Inc., 2012.

[209] Pengfei Xuan, Yueli Zheng, Sapna Sarupria, and Amy W. Apon. SciFlow: A dataflow-driven model architecture for scientific computing using hadoop. In *2013 IEEE International Conference on Big Data*, pages 36–44. IEEE, 2013.

[210] A. Ma and A. R. Dinner. Automatic method for identifying reaction coordinates in complex systems. *J. Phys. Chem. B*, 109:6769–6779, 2005.

[211] W. Li and A. Ma. Recent developments in methods for identifying reaction coordinates. *Mol. Simul.*, 40:784–793, 2014.

[212] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 6 of *Springer Texts in Statistics.* Springer, 2013.

[213] G. Guo, Y. Zhang, and H. Liu. Effect of methane adsorption on the lifetime of a dodecahedral water cluster immersed in liquid water: A molecular dynamics study on the hydrate nucleation mechanisms. *J. Phys. Chem. C*, 111(6):2595–2606, 2007.

[214] A. H. Nguyen and V. Molinero. Identification of clathrate hydrates, hexagonal ice, cubic ice, and liquid water in simulations: The CHILL+ algorithm. *J. Phys. Chem. B*, 119(29):9369–9376, 2014.

[215] Johannes Kästner. Umbrella sampling. *WIRES Comput. Mol. Sci.*, 1:932–942, 2011.

[216] J. S. Hub, B. L. De Groot, and D. Van Der Spoel. g_wham-a free weighted histogram analysis implementation including robust error and autocorrelation estimates. *J. Chem. Theory Comput.*, 6(12):3713–3720, 2010.

[217] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81(8):3684–3690, 1984.

[218] S. Nosé. A molecular dynamics method for simulations in the canonical ensemble. *Mol. Phys.*, 52(2):255–268, 1984.

[219] M. Parrinello and A. Rahman. Polymorphic transitions in single-crystals - a new molecular-dynamics method. *J. Appl. Phys.*, 52(12):7182–7190, 1981.

[220] K. Wm. Hall, S. Carpendale, and P. G. Kusalik. Evidence from mixed hydrate nucleation for a funnel model of crystallization. *Proc. Natl. Acad. Sci. U.S.A.*, page 201610437, 2016.

[221] R. E. Kass and A. E. Raftery. Bayes factors. *J. Am. Stat. Assoc.*, 90(430):773–795, 1995.

[222] C. Liu, Z. Zhang, and G. Guo. Effect of guests on the adsorption interaction between a hydrate cage and guests. *RSC Adv.*, 6(108):106443–106452, 2016.

[223] M. Matsumoto. Four-body cooperativity in hydrophonic association of methane. *J. Phys. Chem. Lett.*, 1(10):1552–1556, 2010.

[224] J. Russo, F. Romano, and H. Tanaka. New metastable form of ice and its role in the homogeneous crystallization of water. *Nat. Mater.*, 13:733–739, 2014.

[225] R. J. Allen, D. Frenkel, and P. R. ten Wolde. Forward flux sampling-type schemes for simulating rare events: Efficiency analysis. *J. Chem. Phys.*, 124:194111, 2006.

[226] R.S. DeFever, W. Hanger, J. Kilgannon, A. W. Apon, L. B. Ngo, and S. Sarupria. Building a scalable forward flux sampling framework using big data and hpc. In *PEARC '19: Proceedings of the Practice and Experience on Advanced Research Computing*, 2019.

[227] H. R. Pruppacher and J. D. Klett. *Microphysics of Clouds and Precipitation.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 2 edition, 1997.

[228] C. Hoose and O. Mohler. Heterogeneous ice nucleation on atmospheric aerosols: a review of results from laboratory experiments. *Atmos. Chem. Phys.*, 12:9817–9854, 2012.

[229] T. S. Van Erp and P. G. Bolhuis. Elaborating transition interface sampling methods. *J. Comput. Phys.*, 205(1):157–181, 2005.

[230] M. A. Rohrdanz, W. Zheng, and C. Clementi. Discovering mountain passes via torchlight: methods for the definition of reaction coordinates and pathways in complex macromolecular reactions. *Ann. Rev. Phys. Chem.*, 64:295–316, 2013.

[231] CK Wensel. Cascading: Defining and executing complex and fault tolerant data processing workflows on a Hadoop cluster, 2015.

[232] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. The Google file system. *ACM SIGOPS operating systems review*, 37(5):29–43, 2003.

[233] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, January 2008.

[234] Kyong-Ha Lee, Yoon-Joon Lee, Hyunsik Choi, Yon Dohn Chung, and Bongki Moon. Parallel data processing with MapReduce: a survey. *AcM sIGMoD Record*, 40(4):11–20, 2012.

[235] S. Pronk, S. Pall, R. Schulz, P. Larsson, P. Bjelkmar, M. R. Apostolov, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, and E. Lindahl. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics*, 29:845–854, 2013.

[236] Sriram Krishnan, Mahidhar Tatineni, and Chaitanya Baru. myhadoop-hadoop-on-demand on traditional hpc resources. *San Diego Supercomputer Center Technical Report TR-2011-2, University of California, San Diego*, 2011.

[237] L. B. Ngo, E. Duffy, and A. Apon. Teaching HDFS/MapReduce systems concepts to undergraduates. In *Proceedings of the NSF/TCPP Workshop on Parallel and Distributed Computing Education*, 2014.

[238] W. C. Moody, L. B. Ngo, E. Duffy, and A. Apon. JUMMP: Job uninterrupted maneuverable mapreduce platform. In *Proceedings of the 2013 IEEE International Conference on Cluster Computing*, 2013.

[239] K. Kratzer, J. T. Berryman, A. Taudt, J. Zeman, and A. Arnold. The flexible rare event sampling harness system (FRESHS). *Comput. Phys. Commun.*, 185(7):1875–1885, 2014.

[240] Rosalind Allen, Juho Lintuvori, and Kevin Stafford. Parallel forward flux sampling, 2013.

[241] R S DeFever and S Sarupria. Contour forward flux sampling: Sampling rare events along multiple collective variables. *J. Chem. Phys.*, 150(2):024103, 2019.

[242] Baron Peters. *Reaction Rate Theory and Rare Events.* Elsevier, 2017.

[243] Phillip L. Geissler, Christoph Dellago, and David Chandler. Kinetic pathways of ion pair dissociation in water. *J. Phys. Chem. B*, 103(18):3706–3710, 1999.

[244] E. Pluhařová, M. D. Baer, G. K. Schenter, P. Jungwirth, and C. J. Mundy. Dependence of the rate of lif ion-pairing on the description of molecular interaction. *J. Phys. Chem. B*, 120(8):1749–1758, 2015.

[245] M. Moqadam, A. Lervik, E. Riccardi, V. Venkatraman, B. K. Alsberg, and T. S. van Erp. Local initiation conditions for water autoionization. *Proc. Natl. Acad. Sci. USA*, 115(20):E4569–E4576, 2018.

[246] W. E, W. Ren, and E. Vanden-Eijnden. String method for the study of rare events. *Phys. Rev. B*, 66:052301, 2002.

[247] L. Maragliano, A. Fischer, E. Vanden-Eijnden, and G. Ciccotti. String method in collective variables: Minimum free energy paths and isocommittor surfaces. *J. Chem. Phys.*, 125(2):024106, 2006.

[248] T. S. van Erp. Reaction rate calculation by parallel path swapping. *Phys. Rev. Lett.*, 98(26):268301, 2007.

[249] P. Tiwary and M. Parrinello. From metadynamics to dynamics. *Phys. Rev. Lett.*, 111(23):230602, 2013.

[250] H. Jung, K. Okazaki, and G. Hummer. Transition path sampling of rare events by shooting from the top. *J. Chem. Phys.*, 147(15):152716, 2017.

[251] F. A. Escobedo, E. E. Borrero, and J. C. Araque. Transition path sampling and forward flux sampling. applications to biological systems. *J. Phys. Condens. Mat.*, 21(33):333101, 2009.

[252] S. B. Kim, C. J. Dsilva, I. G. Kevrekidis, and P. G. Debenedetti. Systematic characterization of protein folding pathways using diffusion maps: Application to trp-cage miniprotein. *J. Chem. Phys.*, 142(8):085101, 2015.

[253] A. Vijaykumar, P. R. ten Wolde, and P. G. Bolhuis. Rate constants for proteins binding to substrates with multiple binding sites using a generalized forward flux sampling expression. *J. Chem. Phys.*, 148(12):124109, 2018.

[254] J. Lin. Divergence measures based on the shannon entropy. *IEEE T. Inform. Theory*, 37(1):145–151, 1991.

[255] C. Leitold, W. Lechner, and C. Dellago. A string reaction coordinate for the folding of a polymer chain. *J. Phys. Condens. Mat.*, 27(19):194126, 2015.

[256] R. B. Best and G. Hummer. Reaction coordinates and rates from transition paths. *Proc. Natl. Acad. Sci. USA*, 102(19):6732–6737, 2005.

[257] A. Singer, R. Erban, I. G. Kevrekidis, and R. R. Coifman. Detecting intrinsic slow variables in stochastic dynamical systems by anisotropic diffusion maps. *Proc. Natl. Acad. Sci. USA*, 106(38):16090–16095, 2009.

[258] W. Lechner, J. Rogal, J. Juraszek, B. Ensing, and P. G. Bolhuis. Nonlinear reaction coordinate analysis in the reweighted path ensemble. *J. Chem. Phys.*, 133(17):174110, 2010.

[259] D. Mendels, G. Piccini, and M. Parrinello. Collective variables from local fluctuations. *J. Phys. Chem. Lett.*, 9(11):2776–2781, 2018.

[260] P. G. Bolhuis, C. Dellago, and D. Chandler. Reaction coordinates of biomolecular isomerization. *Proc. Natl. Acad. Sci. USA*, 97(11):5877–5882, 2000.

[261] A. Warmflash, P. Bhimalapuram, and A. R. Dinner. Umbrella sampling for nonequilibrium processes. *J. Chem. Phys.*, 127(15):154112, 2007.

[262] A. Dickson, A. Warmflash, and A. R. Dinner. Nonequilibrium umbrella sampling in spaces of many order parameters. *J. Chem. Phys.*, 130(7):074104, 2009.

[263] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[264] P. Das, M. Moll, H. Stamati, L. E. Kavraki, and C. Clementi. Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proc. Natl. Acad. Sci.*, 103(26):9885–9890, 2006.

[265] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proc. Natl Acad. Sci. USA*, 102(21):7426–7431, 2005.

[266] A. L. Ferguson, A. Z. Panagiotopoulos, I. G. Kevrekidis, and P. G. Debenedetti. Nonlinear dimensionality reduction in molecular simulation: The diffusion map approach. *Chem. Phys. Lett.*, 509(1):1–11, 2011.

[267] E. Chiavazzo, R. Covino, R. R. Coifman, C. W. Gear, A. S. Georgiou, G. Hummer, and I. G. Kevrekidis. Intrinsic map dynamics exploration for uncharted effective free-energy landscapes. *Proc. Natl. Acad. Sci. USA*, 114(28):E5494–E5503, 2017.

[268] W. Chen and A. L. Ferguson. Molecular enhanced sampling with autoencoders: On-the-fly collective variable discovery and accelerated free energy landscape exploration. *J. Comput. Chem.*, 39(25):2079–2102, 2018.

[269] J. Marcelo L. Ribeiro, P. Bravo, Y. Wang, and P. Tiwary. Reweighted autoencoded variational bayes for enhanced sampling (rave). *J. Chem. Phys.*, 149(7):072301, 2018.

[270] M. M. Sultan and V. S. Pande. Automated design of collective variables using supervised machine learning. *J. Chem. Phys.*, 149(9):094106, 2018.

[271] C. Wehmeyer and F. Noé. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *J. Chem. Phys.*, 148(24):241703, 2018.

[272] E. Vanden-Eijnden and M. Venturoli. Exact rate calculations by trajectory parallelization and tilting. *J. Chem. Phys.*, 131(4):044120, 2009.

[273] E. E. Borrero, M. Weinwurm, and C. Dellago. Optimizing transition interface sampling simulations. *J. Chem. Phys.*, 134(24):244118, 2011.

[274] R.S. DeFever, C. Targonski, S. Hall, M. C. Smith, and S. Sarupria. A generalized deep learning approach for local structure identification in molecular simulations. *Chem. Sci.*, 2019.

[275] Paul J Steinhardt, David R Nelson, and Marco Ronchetti. Bond-orientational order in liquids and glasses. *Phys. Rev. B*, 28(2):784, 1983.

[276] Peter Mahler Larsen, Søren Schmidt, and Jakob Schiøtz. Robust Structural Identification via Polyhedral Template Matching. *Modell. Simul. Mater. Sci. Eng.*, 24(5):055007, 2016.

[277] GJ Ackland and AP Jones. Applications of local crystal structure measures in experiment and simulation. *Phys. Rev. B*, 73(5):054104, 2006.

[278] J Dana Honeycutt and Hans C Andersen. Molecular Dynamics Study of Melting and Freezing of Small Lennard-Jones Clusters. *J. Phys. Chem.*, 91(19):4950–4963, 1987.

[279] W. Lechner and C. Dellago. Accurate determination of crystal structures based on averaged local bond order parameters. *J. Chem. Phys.*, 129(11):114707, 2008.

[280] A. Reinhardt, J. P. K. Doye, E. G. Noya, and C. Vega. Local order parameters for use in driving homogeneous ice nucleation with all-atom models of water. *J. Chem. Phys.*, 137(19):194504, 2012.

[281] A. J. Mukhtyar and F. A. Escobedo. Developing local order parameters for order–disorder transitions from particles to block copolymers: Methodological framework. *Macromolecules*, 51(23):9769–9780, 2018.

[282] W. Chen, A. R. Tan, and A. L. Ferguson. Collective variable discovery and enhanced sampling using autoencoders: Innovations in network architecture and error function design. *J. Chem. Phys.*, 149(7):072312, 2018.

[283] H. Jung, R. Covino, and G. Hummer. Artificial intelligence assists discovery of reaction coordinates and mechanisms from molecular dynamics simulations. *arXiv preprint arXiv:1901.04595*, 2019.

[284] J. Behler and M. Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98(14):146401, 2007.

[285] J. Behler. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Phys. Chem. Chem. Phys.*, 13(40):17930–17955, 2011.

[286] S Chmiela, A Tkatchenko, H E Sauceda, I Poltavsky, K T Schütt, and K Müller. Machine learning of accurate energy-conserving molecular force fields. *Sci. Adv.*, 3(5):e1603015, 2017.

[287] H Chan, M J Cherukara, B Narayanan, T D Loeffler, C Benmore, S K Gray, and S K R S Sankaranarayanan. Machine learning coarse grained models for water. *Nature communications*, 10(1):379, 2019.

[288] W. F. Reinhart, A. W. Long, M. P. Howard, A. L. Ferguson, and A. Z. Panagiotopoulos. Machine learning for autonomous crystal structure identification. *Soft Matter*, 13(27):4733–4745, 2017.

[289] Pablo M Piaggi and Michele Parrinello. Predicting polymorphism in molecular crystals using orientational entropy. *Proc. Natl. Acad. Sci. USA*, 115(41):10251–10256, 2018.

[290] Siva Dasetty, John K Barrows, and Sapna Sarupria. Adsorption of amino acids on graphene: assessment of current force fields. *Soft Matter*, 15(11):2359–2372, 2019.

[291] P. Geiger and C. Dellago. Neural networks for local structure detection in polymorphic systems. *J. Chem. Phys.*, 139(16):164105, 2013.

[292] W. F. Reinhart and A. Z. Panagiotopoulos. Automated crystal characterization with a fast neighborhood graph analysis method. *Soft matter*, 14(29):6083–6089, 2018.

[293] M. Spellings and S. C. Glotzer. Machine learning for crystal identification and discovery. *AIChE J.*, 64(6):2198–2206, 2018.

[294] C Dietz, T Kretz, and MH Thoma. Machine-learning approach for local classification of crystalline structures in multiphase systems. *Phys. Rev. E*, 96(1):011301, 2017.

[295] Maxwell Fulford, Matteo Salvalaglio, and Carla Molteni. DeepIce: a Deep Neural Network Approach to Identify Ice and Water Molecules. *J. Chem. Inf. Model.*, 0(ja):null, 0.

[296] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[297] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[298] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[299] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[300] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[301] MA Carignano, PB Shepson, and I Szleifer*. Molecular dynamics simulations of ice growth from supercooled water. *Mol. Phys.*, 103(21-23):2957–2967, 2005.

[302] S Sarupria and P G Debenedetti. Molecular dynamics study of carbon dioxide hydrate dissociation. *J. Phys. Chem. A*, 115(23):6102–6111, 2011.

[303] V Yamakov, D Wolf, SR Phillpot, AK Mukherjee, and H Gleiter. Deformation-mechanism map for nanocrystalline metals by molecular-dynamics simulation. *Nat. Mater.*, 3(1):43, 2004.

[304] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[305] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shape modeling. In *CVPR*, volume 1, page 3, 2015.

[306] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[307] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[308] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[309] W Lechner, C Dellago, and P G Bolhuis. Reaction coordinates for the crystal nucleation of colloidal suspensions extracted from the reweighted path ensemble. *J. Chem. Phys.*, 135(15):154110, 2011.

[310] P. V. Hobbs. *Ice physics.* Oxford University Press, 2010.

[311] D. T. Limmer and D. Chandler. Premelting, fluctuations, and coarse-graining of water-ice interfaces. *J. Chem. Phys.*, 141(18):18C505, 2014.

[312] Tanja Kling, Felix Kling, and Davide Donadio. Structure and Dynamics of the Quasi-Liquid Layer at the Surface of Ice from Molecular Simulations. *J. Phys. Chem. C*, 122(43):24780–24787, 2018.

[313] M. A. Boles, M. Engel, and D. V. Talapin. Self-assembly of colloidal nanocrystals: From intricate structures to functional materials. *Chem. Rev.*, 116(18):11220–11289, 2016.

[314] K. Thorkelsson, P. Bai, and T. Xu. Self-assembly and applications of anisotropic nanomaterials: A review. *Nano Today*, 10(1):48–66, 2015.

[315] A. Kumar and V. Molinero. Why is gyroid more difficult to nucleate from disordered liquids than lamellar and hexagonal mesophases? *J. Phys. Chem. B*, 122(17):4758–4770, 2018.

[316] Ankita J Mukhtyar and Fernando A Escobedo. Developing local order parameters for order–disorder transitions from particles to block copolymers: Application to macromolecular systems. *Macromolecules*, 51(23):9781–9788, 2018.

[317] H Acharya, S Vembanur, S N Jamadagni, and S Garde. Mapping hydrophobicity at the nanoscale: Applications to heterogeneous surfaces and proteins. *Faraday Discuss.*, 146:353–365, 2010.

[318] Amish J Patel and Shekhar Garde. Efficient Method to Characterize the Context-Dependent Hydrophobicity of Proteins. *J. Phys. Chem. B*, 118(6):1564–1573, 2014.

[319] E. Xi, V. Venkateshwaran, L. Li, N. Rego, A. J. Patel, and S. Garde. Hydrophobicity of proteins and nanostructured solutes is governed by topographical and chemical context. *Proc. Natl. Acad. Sci.*, 114(51):13345–13350, 2017.

[320] Z. Jiang, R. C. Remsing, N. B. Rego, and A. J. Patel. Characterizing solvent density fluctuations in dynamical observation volumes. *J. Phys. Chem. B*, 2019.

[321] A J Patel, P Varilly, D Chandler, and S Garde. Quantifying Density Fluctuations in Volumes of All Shapes and Sizes Using Indirect Umbrella Sampling. *J. Stat. Phys.*, 145(2):265–275, 2011.

[322] J R Morris and X Song. The melting lines of model systems calculated from coexistence simulations. *J. Chem. Phys.*, 116(21):9352–9358, 2002.

[323] M Zaharia, M Chowdhury, M J Franklin, S Shenker, and I Stoica. Spark: Cluster computing with working sets. *HotCloud*, 10(10-10):95, 2010.

[324] P. G. Bolhuis. Rare events via multiple reaction channels sampled by path replica exchange. *J. Chem. Phys.*, 129(11):114108, 2008.

[325] O Melnikov, V Sarvanov, RI Tyshkevich, Vladimir Yemelichev, and Igor E Zverovich. *Exercises in graph theory*, volume 19 of *Texts in the Mathematical Sciences*. Springer Science & Business Media, 2013.

[326] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1:19–25, 2015.

[327] V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, and C. Simmerling. Comparison of multiple amber force fields and development of improved protein backbone parameters. *Proteins*, 65(3):712–725, 2006.

[328] B. Hess, H. Bekker, H. J. C. Berendsen, and J. G. E. M. Fraaije. Lincs: a linear constraint solver for molecular simulations. *J. Comput. Chem.*, 18(12):1463–1472, 1997.

[329] B. Hess. P-LINCS: A parallel linear constraint solver for molecular simulation. *J. Chem. Theory Comput.*, 4(1):116–122, 2008.

[330] G. Bussi, D. Donadio, and M. Parrinello. Canonical sampling through velocity rescaling. *J. Chem. Phys.*, 126(1):014101, 2007.

[331] J. L. F. Abascal, E. Sanz, R. García Fernández, and C. Vega. A potential model for the study of ices and amorphous water: TIP4P/Ice. *J. Chem. Phys.*, 122(23):234511, 2005.

[332] Loup Verlet. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Phys. Rev.*, 159(1):98, 1967.

[333] Tom Darden, Darrin York, and Lee Pedersen. Particle mesh Ewald: An N log (N) method for Ewald sums in large systems. *J. Chem. Phys.*, 98(12):10089–10092, 1993.

[334] S. Miyamoto and P. A. Kollman. Settle: An analytical version of the shake and rattle algorithm for rigid water models. *J. Comput. Chem.*, 13(8):952–962, 1992.

[335] Junmei Wang, Romain M Wolf, James W Caldwell, Peter A Kollman, and David A Case. Development and testing of a general amber force field. *J. Comput. Chem.*, 25(9):1157–1174, 2004.

[336] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen. A smooth particle mesh ewald method. *J. Chem. Phys.*, 103:8577, 1995.

[337] K Lindorff-Larsen, S Piana, K Palmo, P Maragakis, J L Klepeis, R O Dror, and D E Shaw. Improved side-chain torsion potentials for the amber ff99sb protein force field. *Proteins: Structure, Function, and Bioinformatics*, 78(8):1950–1958, 2010.