An Off-Lattice Kinetic Monte Carlo Framework For Long-Time Atomistic Simulations

by

Julien L. Luzzatto

Submitted to the Center for Computational Science and Engineering
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
in
COMPUTATIONAL SCIENCE AND ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2023

Authored by:	Julien L. Luzzatto
	Center for Computational Science and Engineering
	May 22, 2023
Certified by:	Nicolas G. Hadjiconstantinou
	Professor of Mechanical Engineering
	Thesis Supervisor
Accepted by:	Youssef Marzouk
	Professor of Aeronautics and Astronautics
	Co-Director, Center for Computational Science and Engineering

An Off-Lattice Kinetic Monte Carlo Framework For Long-Time Atomistic Simulations

by

Julien L. Luzzatto

Submitted to the Center for Computational Science and Engineering
on May 22, 2023, in partial fulfillment of the
requirements for the degree of
Master of Science in Computational Science and Engineering

Abstract

The goal of this thesis is to develop an off-lattice Kinetic Monte Carlo (KMC) framework to simulate the atomistic dynamics of materials at extreme conditions over long time scales. Despite the dramatic increase in computational power over the last few decades, rigorous approaches such as classical Molecular Dynamics (MD) techniques cannot access the engineering and experimental time scales due to the fundamental scaling limitation constrained by atomic vibrations. KMC approaches are powerful stochastic computational techniques that focus on the simulation of rare atomistic events in order to analyze the coarse-grained dynamics of condensed matter systems and replicate non-equilibrium phenomena in a statistical fashion. However, their application to problems at extreme conditions — such as those encountered in materials science under high pressure, temperature, and radiation — has been limited by the complexity of atomistic interactions, by the variability and instability of underlying structures, and by the computational cost of simulating large systems over sufficiently long time-scales.

To address such challenges, this thesis proposes an off-lattice, modular and scalable KMC framework that features adaptive inferred structures, efficient process sampling and dynamic rate constant calculations, together with the corresponding Julia implementation. The developed KMC framework is justified theoretically, described step-by-step methodologically, and then validated against MD results for early-time dynamics.

Thesis Supervisor: Nicolas G. Hadjiconstantinou
Title: Professor of Mechanical Engineering

Acknowledgments

I would like to express my sincere gratitude to my advisor, Professor Nicolas Hadjiconstantinou for his continued guidance, support and advice throughout my two years at MIT.

I would also like to thank Professor Youssef Marzouk and the Uncertainty Quantification group for the helpful discussions and for the precious feedback over the last two years.

Many thanks to my friends and collaborators within the Center for the Exascale Simulation of Materials in Extreme Environments (CESMIX) for the joint efforts and the insightful discussions. In particular, special thanks to Dr Mathew Swisher — who contributed significantly to the early simulation efforts and performed most of the molecular dynamics simulations — and Dionysios Sema, for the thoughtful suggestions.

Finally, I am very grateful to my parents, family and friends for their unconditional support.

Contents

1 Introduction **9**

1.1 Motivation 9

1.2 Objectives 11

1.3 Outline 11

2 Theoretical Framework **13**

2.1 Molecular Dynamics 13

2.2 Rare Events Simulation and Coarse Graining 18

2.3 Kinetic Monte Carlo 21

3 Methodology **29**

3.1 Off-Lattice Kinetic Monte Carlo 29

3.2 Dynamic Calculations of Rate Constants 34

3.3 Implementation 45

4 Results **49**

4.1 Molecular Dynamics Simulation of Oxidation 49

4.2 Kinetic Monte Carlo Simulation of Oxidation 51

4.3 Validation 54

5 Conclusion **61**

Bibliography 63

Chapter 1

Introduction

1.1 Motivation

Over the last few decades, the unprecedented increase in computing power enabled by modern supercomputers has transformed the fields of computational chemistry, computational physics and materials science by allowing simulations at unparalleled fidelity and scale.

These communities have flourished by focusing on a relatively limited variety of foundational computational techniques, such as density functional theory (DFT) [ED11, DG12] and molecular dynamics (MD) [HJMM93, BM96]. Such methods have been successfully and efficiently adapted to run on high performance, parallel platforms, but feature intrinsic limitations. Most of the progress enabled by the increasing availability of computing power has been steered towards weak-scaling, meaning that a greater number of processors permits to simulate larger problems. In other words, thanks to larger and larger computers, we are now able to simulate larger and larger systems, but are still severely limited in terms of increasing the physical time simulated. Ideally, computational advances should apply equally to system time-scales, length-scales (system sizes), or accuracy [PPV20].

Although computational processing gains are always welcome, computational gains in parallel processing are typically hard to translate into longer time integration, which is difficult to parallelize. This is a rather important consideration for a wide range on

practical problems of interest which feature a large gap between the "outer", mesoscopic, or possibly macroscopic, time-scale of interest and the "inner", microscopic time-scale at which these algorithms operate (which in most cases is an integration timestep associated with atomistic vibrations). Examples of such problems include heterogeneous catalysis [APR19] and materials degradation [RS06].

In this context, Kinetic Monte Carlo (KMC) [Gil77, Gil07] techniques have emerged as a powerful method for bridging this time-scale gap. The core idea behind Kinetic Monte Carlo methods is to adopt a coarse-grained approach, which post-processes the information it receives from higher-fidelity methods such as DFT and MD, with the goal of describing only rare events that govern the (relatively) slow conformational changes (from 10^{-9} s to s, see figure 1-1).

In this thesis we will describe the theoretical and mathematical framework governing Kinetic Monte Carlo algorithms and analyse how it can find its place in a hierarchical multi-scale modeling approach, by focusing on a material system of current practical interest.

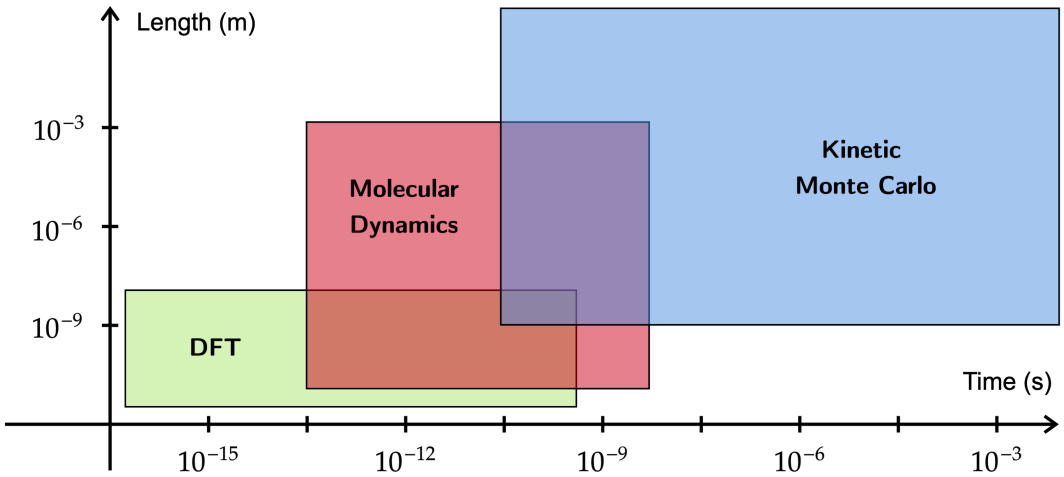

Figure 1-1: The time-scale problem: DFT and MD cannot access time scales much larger than those of atomic vibrations; KMC can simulate longer times, but involves potentially large approximations of the individual atomistic dynamics (adapted from [Kra09]).

1.2 Objectives

The MIT Center for the Exascale Simulation of Materials in Extreme Environments (CESMIX) is focused on the creation of the next generation of simulation tools performing predictive simulations of the degradation of complex materials under extreme loading (within the context of supersonic vehicles, for instance), which are inaccessible to direct experimental observation. Among the main objectives, we would like to develop ab initio computational methods for modeling such materials at high temperatures, and design inference-based methods for coarse-graining quantum mechanical interaction potentials.

One of the specific objectives of this research is to develop atomistic simulations to study the oxidation of materials at long time scales, which requires bridging the gap between the quantum mechanical and experimental time scales. In this thesis, we perform the first steps towards addressing such challenges by developing a dynamic Kinetic Monte Carlo framework which reproduces and extrapolates molecular dynamics results in an accelerated and coarse-grained fashion, in the context of a specific material of interest, namely hafnium and its oxidation.

1.3 Outline

The thesis outline is as follows. In Chapter 2, we will first motivate and detail our Molecular Dynamics approach to the problem of interest. We will then discuss possible coarse-graining approaches and methods to isolate rare events, and finally introduce the mathematical framework shaping the Kinetic Monte Carlo approach.

In Chapter 3 we describe the methodology that we have designed to tackle our specific problem of interest using a Kinetic Monte Carlo framework. We start by describing the mechanics of the off-lattice Kinetic Monte Carlo that we have constructed. We will then detail our approach for dynamically evaluating the rate constants associated with the coarse-grained system description. Finally, we will introduce the Julia implementation.

Chapter 4 includes the results of our simulations. We will present long-time Kinetic Monte Carlo simulations of hafnium oxidation and validate its key features for early times by comparing them to those of Molecular Dynamics simulations. We also discuss a series of ideas for further extensions and improvements.

We will conclude in Chapter 5 by summarizing our results and outlining future work suggestions.

Chapter 2

Theoretical Framework

2.1 Molecular Dynamics

Molecular Dynamics (MD) [AY20] is a computational approach for studying the motion and interactions of sets of atoms and molecules. Given a specific interatomic potential (IAP), the core idea is to generate the individual atomic trajectories of the particles comprising the system of interest by integrating numerically Newton's equations of motion. Overall, molecular dynamics is a powerful tool for understanding the dynamic behavior of complex systems at the atomic and molecular scale, and has applications in many fields, including materials science, chemistry and biophysics [Li05].

Let us consider a system with N atoms in a volume $\mathbf{\Omega}$, yielding a $3N$-dimensional configuration space $\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$. To initialize a standard MD simulation algorithm, we define the initial conditions in terms of particle positions and velocities, which in turn determine the macroscopic quantities of the system, such as pressure and temperature. The total energy of the system can be written as the sum of the kinetic and potential energy:

$$E \equiv K + V = \sum_{i=1}^{N} \frac{1}{2} m_i |\dot{\mathbf{r}}_i(t)|^2 + V(\mathcal{R}(\mathbf{t})) \tag{2.1}$$

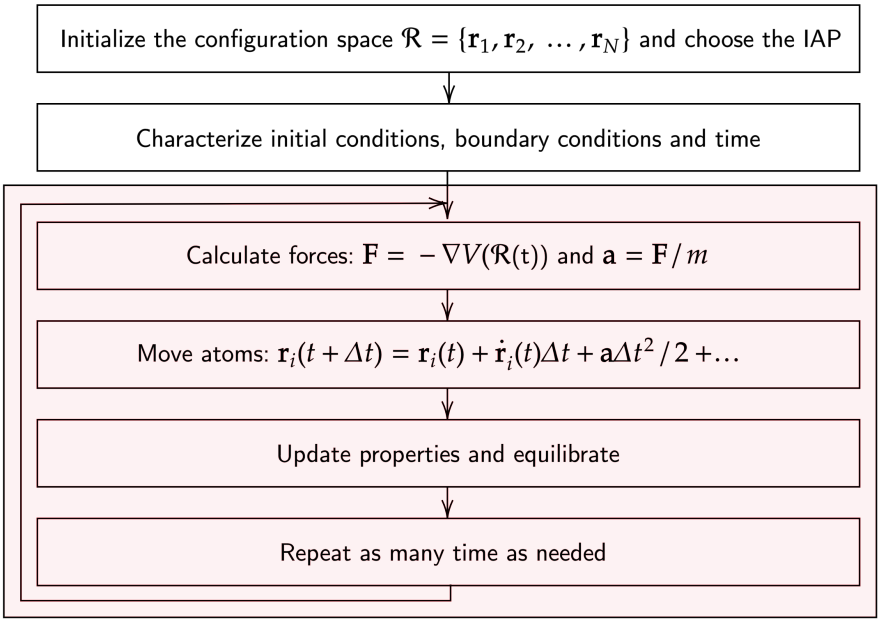

Figure 2-1: Schematic flowchart of a Molecular Dynamics algorithm.

Among the essential components of a molecular dynamics calculation, we highlight the importance of an appropriate initialization, of an efficient parallelization, and of force calculations. A flowchart of the algorithm is also illustrated in figure 2-1.

As we have mentioned previously, the initialization of the simulation includes the definition of the initial conditions and of the boundary conditions. The initial structure at the start of the simulation clearly depends on the problem of interest, and can have large implications on the early-time dynamics. For crystalline solids — such as the hafnium bulk we will study in further chapters — the generation of the structure is usually straightforward, and simply requires the knowledge of key material-specific properties, such as bond lengths, lattice constants and coordination numbers. For liquids or for amorphous materials, such a generation can be more challenging. Once one defines appropriately the initial positions, characterizing the configuration space $\mathcal{R}(t=0)$, one also defines initial velocities. In the absence of more precise information, these are typically initialized from the equilibrium distribution parameterized by the simulation temperature. According to the equipartition theorem [Li05], this can be done by drawing each component of the $3N$-dimensional space $\dot{\mathcal{R}}(t=0)$ from a Gaussian (Maxwell-Boltzmann, Normal) distribution $\mathcal{N}(0, k_B T/m_i)$, where m_i is the

mass of the i-th particle. The boundary conditions are usually set to be periodic for a bulk system (or periodic in some directions but not others for a system comprising a material slab featuring an interface).

The treatment of the system's boundaries also plays an important role in the parallelization of molecular dynamics algorithms, in which different parts of the system are assigned to different computing cores for processing [PPV20]. To do so, one discretizes space and assigns each atom or molecule to a given core, and then perform periodic checks to prevent conflicts and account for crossings between different sections. As long as the communication cost (or delay) is much smaller than the computational gain obtained from increasing the number of cores, one can use this process to simulate progressively larger systems: the speedup resulting from this process is quantified by the weak-scaling that we referred to in the introduction 1.3. Increasing the number of cores usually helps distribute calculations among more processing units and thus treat more atoms, but cannot help accelerate the calculation of forces acting on individual atoms: this is the main reason molecular dynamics techniques are said to perform poorly in terms of strong-scaling.

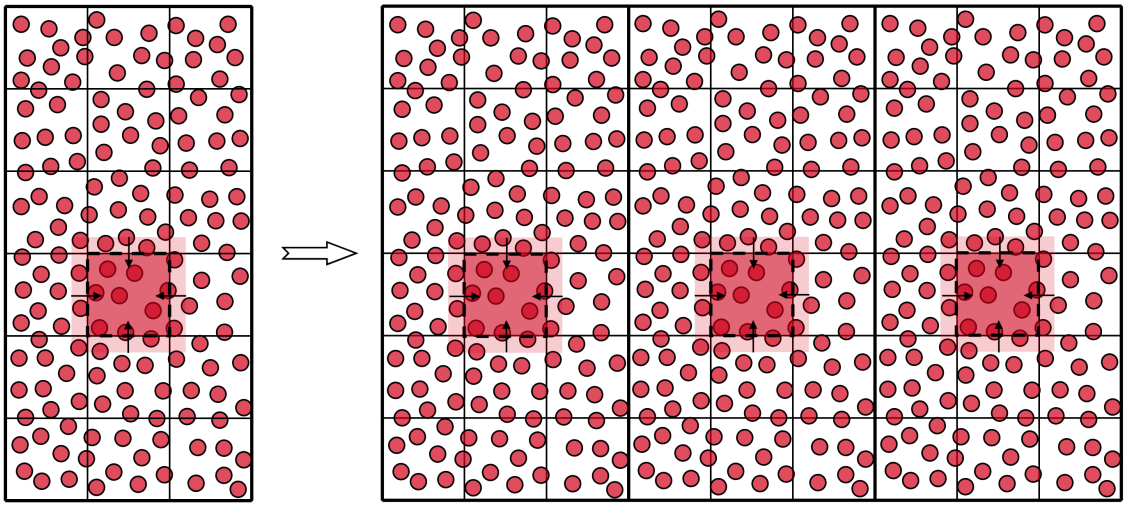

Figure 2-2: Molecular Dynamics techniques perform well in terms of weak-scaling: by increasing the number of cores, we can increase the system's size (adapted from [PPV20])

Force calculations lie at the core of molecular dynamics, since an overwhelming majority of the computational time is dedicated to integrate Newton's equation to obtain those forces. In the context of the Born-Oppenheimer approximation [CDS81], which considers the nuclei to be fixed when solving for the electronic structure, we can predict molecular motion by solving:

$$m_i \ddot{\mathbf{r}}_i(t) = \mathbf{F}_i \equiv -\nabla V(\mathbf{r}_i(t)) \tag{2.2}$$

where $\mathbf{r}_i(t)$ describes the position of the i-th particle at time t.

Several methods exist for integrating the equations of motion, exemplified by the Verlet algorithm [Ver67]:

$$\mathbf{r}_i(t_0 + \Delta t) = 2\mathbf{r}_i(t_0) - \mathbf{r}_i(t_0 - \Delta t) + \ddot{\mathbf{r}}_i(t_0)(\Delta t)^2 + \mathcal{O}((\Delta t)^4) \tag{2.3}$$

$$= 2\mathbf{r}_i(t_0) - \mathbf{r}_i(t_0 - \Delta t) + (\mathbf{f}_i(t_0)/m_i)\,(\Delta t)^2 + \mathcal{O}((\Delta t)^4) \tag{2.4}$$

a fourth-order (for positions [AT17]), fixed-time-step integrator (the time step, Δt remains constant throughout the simulation). This algorithm has served as one of the workhorses in the field due to its ability of combine extreme simplicity, good numerical stability, good accuracy and good conservation properties [AT17], provided that Δt is small enough.

The interatomic potential has a large effect on the computational cost of the simulations as well as the fidelity of the computational results. From an uncertainty quantification perspective, it is also worth noting that molecular dynamics trajectories inherit all intrinsic approximations and errors carried by the potential [BKC12]. In other words, the accuracy of interatomic potentials is a critical ingredient of atomistic simulations. Developing accurate potentials is a complex, time-consuming effort which requires a judicious combination of experimental data, theoretical calculations, as well as considerable domain knowledge.

Due to the complexity and diversity of types of interatomic interactions, interatomic potentials need to be tailored to the material or class of material of interest,

while in many cases only a subset of properties of the material of interest can be captured accurately (e.g. thermal versus mechanical versus electrical). As a result, a number of such potentials exists. Common types of interatomic potentials include classical pair potentials [Jon24], which describe the interactions between pairs of atoms, and more and more complex — and computationally expensive — potentials, such as embedded atom models [DB84], which takes into account the electron density, bond order potentials [SB12], accounting for the strength of chemical bonds, or modern machine-learning potentials [BP07]. The CESMIX project in part concerns itself with the task of designing an accurate and efficient interatomic potential for modeling the physical, thermal and oxidation properties of hafnium at high temperatures [NR23]. Beyond addressing the immediate modeling need associated with this project, this effort will be valuable in answering the question whether training a large amount of diverse and even suitably chosen data using machine learning techniques can replace the domain expertise that is so central to the success of potentials for complex materials and processes.

In this thesis, we also use the Charge-Optimized Many-Body (COMB) empirical potential [RSB94]. The COMB potential is part of the class of variable charge potentials: besides pairwise interactions between atoms, the COMB potential takes into account many-body interactions, including electrostatic and polarization effects, to more accurately capture the behavior of complex materials. To do so, it relies on an amalgamation of quantum mechanical and classical approaches. Specifically, the potential energy of a given system is calculated using a combination of ab initio electronic structure calculations — which provide insights on the charge distribution and the bonding in the material — and empirical fitting to experimental observations and data — which essentially fits the potential to match material properties such as lattice constants and elastic properties. The potential energy of a system reads:

$$E_P = \sum_i E_i^{\text{self}}(q_i) + E^{\text{barr}}(q_i) + \sum_{j>i} E_{ij}^{\text{short}}(r_{ij}, q_i, q_j) + E_{ij}^{\text{Coul}}(r_{ij}, q_i, q_j) + E^{\text{corr}}(r_{ij}, \theta_{ij}) \tag{2.5}$$

where E_i^{self} is the self-energy of atom i, including atomic ionization energies and the

descriptors of the electronic structure, E^{barr} is a charge barrier function, E_{ij}^{self} is the bond-order potential between atoms i and j, E_{ij}^{Coul} is the Coulomb interaction, and E^{corr} are angular correction terms [RSB94].

Overall, COMB is a relatively expensive potential in computational terms, due to the inclusion of many-body interactions and electrostatics. However, it is generally considered to be quite accurate for complex materials featuring several many-body interactions, and features the advantage of accurately capturing the behaviour of solid-state materials — and especially hafnium [SDK$^+$10] — over a wide range of temperature and pressure conditions, making it a valuable tool for simulating systems in extreme conditions or in (quasi) non-equilibrium states.

We will present the results of our molecular dynamics simulations for our problem of interest in section 4.1.

2.2 Rare Events Simulation and Coarse Graining

Molecular dynamics methods have enjoyed considerable success over the last few decades. Their simplicity and robustness have made this method a workhorse in chemistry, materials science and computational physics, and are widely used to gain key insights on the structure, dynamics, and thermodynamics of complex material systems. As we have suggested in the motivation, section 1.1, however, a fundamental limitation exists: accessing macroscopic — or even mesoscopic — timescales is prohibitively expensive.

This limitation arises from the nature of the atomistic dynamics of such complex material systems, which feature rare events that follow from overcoming high activation barriers, typically much larger than $k_B T$ [MS05, APR19]. While molecular dynamics techniques resolve atomic vibrations in energetic basins accurately (using a much smaller time step, of the order of the femtosecond, 10^{-15} s), such rare events typically occur on time scales which four or more orders of magnitude larger (of the order of the 10s or 100s of picoseconds, $\sim 10^{-10}$ s) [Vot07]. As a consequence, molecular dynamics simulation trajectories spend the majority of time in the vicinity of

local energy minima without ever accessing the time scales of the high-barrier elementary processes of interest. This discrepancy poses severe constraints to the usefulness of molecular dynamics methods when it comes to studying the dynamics of systems featuring such high-barrier processes.

To overcome this time-scale problem, it is worth noting that such a bottleneck suggests its own solution. Namely, the overwhelming majority of moves is not significant from a mesoscopic perspective, because the system essentially oscillates around local minima without meaningful dynamics on a larger scale, which are driven by the aforementioned rare events. This leads us to the definition of metastates and metastability. From a mesoscopic perspective, the potential energy surface (PES) of a system can be described as a collection of metastates (usually represented by individual energy basins), which are separated by high-energy barriers.

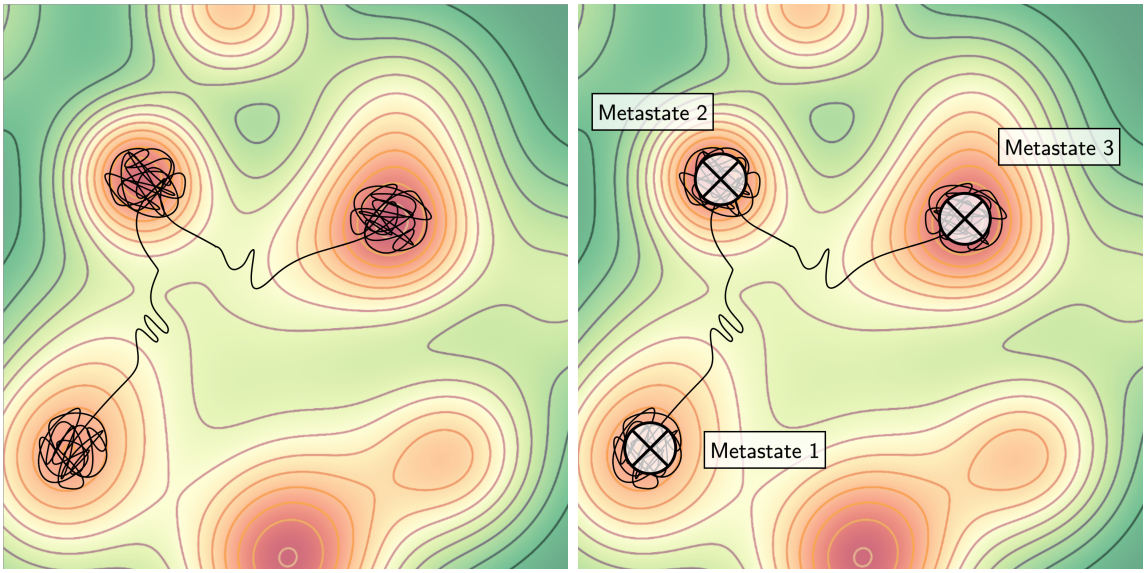

Figure 2-3: Two-dimensional projection of a molecular dynamics trajectory on a potential energy surface and graphical intuition of the characterization of metastates.

A given atom on such PES will likely remain trapped in a metastable state for a significant period of time, before gaining enough energy to escape it, and ending up in another metastate [UMGV05], as illustrated in figure 2-3.

These concepts enable the abstraction of molecular dynamics trajectories onto coarse-grained chains transitioning from one metastate to another, rather than considering the details of the behavior of the particles while they are trapped in a given metastate. In other words, rather than following the trajectory through every vibrational period, we can consider these state-to-state transitions directly.

Moreover, it is intuitively reasonable to assume that after a certain amount of time spent in a given metastate (inside a given energy basin), the future behaviour of the system will not be affected by the preceding state-to-state trajectory, but simply dictated by the local topology of the metastate. In other words, the probability of a system transitioning from one state to another features the Markovian property, and the state-to-state dynamics is simply a Markov chain [VK11]. From the perspective of Transition State Theory [CN17], such probability of a system overcoming a high-energy barrier and transitioning from one metastate i to another metastate j is described by a rate constant [TGK96] that we denote by k_{ij}. Hence, a comprehensive description of the state-to-state dynamics is provided by the following Markovian master equation for each particle i [Jan12]:

$$\frac{\mathrm{d}P_i(t)}{\mathrm{d}dt} = -\sum_{j \neq i} k_{ij} P_i(t) + \sum_{j \neq i} k_{ji} P_j(t) \tag{2.6}$$

Although conceptually and mathematically simple, this system of coupled differential equations can become prohibitively expensive to simulate if not approached properly [APR19]: as an example, let us consider a simple on-lattice representation of a bulk of hafnium oxide with a hundred hafnium atoms and two hundred oxygen atoms, and let us restrict ourselves to the study of the dynamics of the oxygen atoms. Each oxygen site can assume two occupation states (free or occupied), meaning that there are 2^{200} possible configurations in such a simple system, and that the full transition matrix contains $(2^{200})^2 \approx 2.58 \times 10^{120}$ entries. In other words, formulations which avoid evaluation and manipulation of the complete transition matrices are key to tractability.

2.3 Kinetic Monte Carlo

In the last few decades, Monte Carlo methods — taking advantage of random numbers to explore vast configuration spaces and tackle complex problems in a statistically accurate fashion — have been very successful in predicting equilibrium properties [MU49, BH10]. An ingenious idea to address the high-dimensional Markovian problem 2.6 is to use Monte Carlo methods to approximate dynamical properties [Vot07].

The core idea behind a Kinetic Monte Carlo (KMC) algorithm is to generate individual state-to-state trajectories in a stochastic fashion, in order to obtain statistically accurate trajectories, and thus dynamical properties, without calculating the entire transition matrix. Such a state-to-state dynamic corresponds in mathematical terms to a Markov walk. Note that by displacing atoms or groups of atoms from one metastate to another, we are transitioning the entire system from a previous state to an updated state.

Operationally, a KMC method needs to perform three crucial tasks at every step. First, for a given configuration, it needs to enumerate accurately the possible transitions. Second, it has to pair to every possible transition — that we can also call process — a rate constant that accounts for the relative likelihood of that specific process. Third, it must sample a process, execute it, and update the time of the simulation in an appropriate fashion, precisely according to the number of available processes and to their relative probability, embodied by the rate constant.

2.3.1 Enumerating Processes

Addressing the first challenge amounts to finding an appropriate balance between accuracy and efficiency. Theoretically speaking, in an unconstrained Markov chain, the system could transition from any state to any other state. In practice, as we will justify in our discussion regarding rate constants, the only realistic transitions occur between states neighboring each other on the potential energy surface. Hence, one can construct the list of available processes by enumerating the available transitions between neighbor sites. Note that to do so, one needs to characterize the concept of

neighborhood. For most of the systems that are tackled with KMC methods, it is possible to map the problem onto a lattice [Reu11] — by exploiting the crystalline structure of the material of interest, for example — which makes it straightforward to define neighborhood around a particular site. As we will explain in section 3.1, however, this method was not viable for our problem of interest, due to the evolution of the crystalline structure during the oxidation process. The characterization of neighborhoods is particularly important from an uncertainty quantification perspective too: miscalculating the number of available processes for a given initial state can lead to significant errors in the time inference.

2.3.2 Calculating Rate Constants

To tackle the second problem of calculating rate constants, one summons Transition State Theory (TST) [Eyr35, TG80], which is the theoretical framework addressing the calculation of rates of chemical reactions.

The fundamental assumption of TST is that there exists a hypersurface in phase space which divides the space into reactants and products, and that the trajectories passing through such surface — going from the reactants to the products — will not cross the surface again before being captured in a product state (dynamical bottleneck assumption). This hypersurface acts as a high-energy intermediate state, called the transition state.

The transition state is characterized by the atomistic geometries and electronic configurations of the system's atoms, which determine its stability and reactivity. For activated processes, it represents the point of highest energy along the reaction pathway, and the rate of a chemical reaction is determined by the energy barrier that separates the reactants from the transition state. This energy barrier is called the activation energy, and it is the energy that must be supplied to the reactants in order to reach the transition state and initiate the reaction. The rate of a chemical reaction, corresponding to the rate constant that we are interested in in our KMC framework,

is then calculated using the Arrhenius equation:

$$k_{ij}^{\mathrm{TST}} = A \exp\left(-\frac{\Delta E_{ij}}{k_B T}\right) \tag{2.7}$$

where k_{ij}^{TST} is the rate constant for an activated process between states i and j, A is the pre-exponential factor that depends on the details of the reaction mechanism, ΔE_{ij} is the energy barrier between states i and j, k_B is Boltzmann's constant, and T is the temperature.

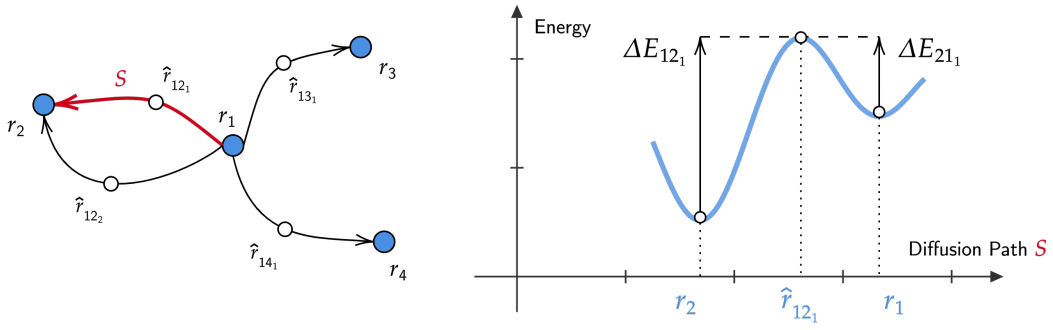

Figure 2-4: Diffusion paths between metastates (for all i, j and k, r_i are configuration states, $\hat{r}_{ij_k}$ are saddle points between state i and state j, ΔE_{ij_k} is the energy barrier from i to j going through saddle point k) and corresponding energy barrier (adapted from [KSDK18]).

As can be seen from equation 2.7, rate constants depend linearly on the TST pre-factor, and exponentially on the energy barrier. As a consequence, coarser approximations are usually adopted when considering pre-factors compared to energy barriers. The predominant resource is the harmonic approximation [TGK96], which is a simplifying assumption according to which the potential energy surface around the transition state can be approximated by a quadratic potential energy function. This allows one to treat atomic vibrations as a collection of independent harmonic oscillators, each with a characteristic frequency and energy, and reduces greatly the number of vibrational modes that need to be considered. Instead of dealing with the complex anharmonic vibrations that may characterize the transition state, the

harmonic approximation assumes that the vibrational motions can be described by a set of normal modes that obey the laws of classical mechanics [XH08]. Such an assumption is only valid when the vibrational motions of the atoms in the transition state are small compared to the vibrational motions of the atoms at the local minimum, or — in other words — if the energy barriers are much larger than $k_B T$. Our calculations show that this assumption is reasonable for hafnium oxidation at high temperatures ($k_B T \approx 0.2$eV, whereas the average energy barriers between metastates were of the order of 3eV).

In the harmonic approximation of transition state theory, the rate constants can be calculated with the so-called Eyring equation [Lai87]:

$$k_{ij}^{\mathrm{hTST}} = \frac{q_{TS}^{\mathrm{vib}}}{q_i^{\mathrm{vib}}} \frac{k_B T}{h} \exp\left(-\frac{\Delta E_{ij}}{k_B T}\right) \tag{2.8}$$

where q_{TS}^{vib} is the vibrational partition function at the transition state, q_i^{vib} is the vibrational partition function at the initial state, and h is Planck's constant. In the context of harmonic TST, the ratio of the partition functions is generally fitted to a constant value characteristic of the process studied [APR19]:

$$\frac{q_{TS}^{\mathrm{vib}}}{q_i^{\mathrm{vib}}} \equiv k_0 \tag{2.9}$$

to avoid the considerable cost of vibrational calculations.

An important class of processes for which these considerations do not apply is non-activated adsorption, also called physisorption. The rate constant for adsorption of a species i onto a free site f is related to the kinetic impingement rate onto the whole surface unit-cell, and the fraction of incoming particles which actually adsorb on a a given free site is governed by a sticking coefficient $S_{i,f}(T)$ [RS06]:

$$k_{i,f}^{\mathrm{ads}}(T, p_i) = S_{i,f}(T) \frac{p_i A_{\mathrm{UC}}}{\sqrt{2\pi m_i k_B T}} \tag{2.10}$$

Here, $k_{i,f}^{\mathrm{ads}}$ is the rate constant for a non-activated adsorption process of a species i onto a free site f on a unit cell of area A_{UC}, p_i is the partial pressure of species i, m_i

is the mass of species i, and $S_{i,f}(T)$ is the sticking coefficient introduced above.

The sticking coefficient can be estimated in several ways. For a majority of non-activated processes, it can be approximated to 1: a particle impinging on the surface at a high-velocity at the location of an adsorption site will always stick, provided that the adsorption site is not occupied (and thus that such a process exists, from a KMC perspective). For more complicated adsorption dynamics, where the velocity of the incoming particles is not high-enough to overcome the barrier along the trajectory to the surface, the process becomes activated, and the sticking coefficient can be related to an activated transition. In other words, within the TST framework, the sticking coefficient can be expressed as [RS06]:

$$S_{i,f}(T) = \frac{q_{TS}^{\text{vib}}}{q_{i,\text{gas}}^{\text{trans}} q_{i,\text{gas}}^{\text{int}}} \frac{k_B T}{h} \frac{A_{i,f}}{A_{\text{UC}}} \exp\left(-\frac{\Delta E_{i,f}^{\text{ads}}}{k_B T}\right) \tag{2.11}$$

where $q_{i,\text{gas}}^{\text{trans}}$ is the two-dimensional translational partition function over the surface unit cell, $q_{i,\text{gas}}^{\text{int}}$ is the partition function of the internal degrees of freedom of the free gas phase, $A_{i,f}$ is the surface area of the free adsorption site f for species i, and $\Delta E_{i,f}^{\text{ads}}$ is the energy barrier along the minimum energy path connecting the gas phase and the adsorption site.

2.3.3 Updating Time and Sampling Processes

A KMC algorithm has to reproduce the time evolution of the dynamics in an appropriate manner despite the coarse graining, precisely because of the time-scale separation between vibrations and translations, and by taking advantage of the knowledge of the rate constants.

Let us fix a given state A in our configuration space, and let us consider a neighbor, accessible state B. The rate constant characterizing the translation of the system from A to B is described by k_{AB}, as detailed in the previous sections. What does k_{AB} represent? Mathematically speaking, k_{AB}^{-1} describes the average time at which the system is expected to transition from state A to state B. Because the probability of the translation is constant for each fraction of time, the probability that the system

has not transitioned after a time τ follows a Poisson distribution:

$$P^{\text{Pois}}(i = 0, k_{AB}\tau) = \frac{(k_{AB}\tau)^i \exp(-k_{AB}\tau)}{i!} = \exp(-k_{AB}\tau) \tag{2.12}$$

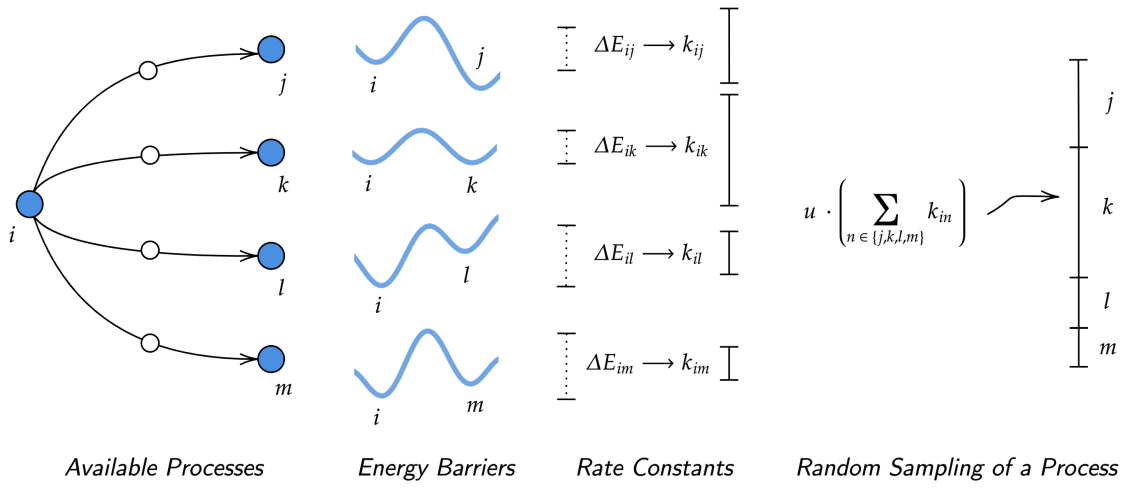

Figure 2-5: Schematic representation of the sampling of a process by weighing the energy barriers.

To calculate the probability that a transition does occur by the time $\tau + \Delta\tau$, we can take the product of the probability that no reaction occurred in the time interval τ times the probability that a reaction occurs in the time interval $(\tau, \Delta\tau)$, which is $\Delta\tau / k_{AB}^{-1} = \Delta\tau k_{AB}$. Hence, we have:

$$P^{\text{Pois}}(i > 0, k_{AB}\tau)\Delta\tau = k_{AB} \exp(-k_{AB}\tau)\Delta\tau \tag{2.13}$$

$$P^{\text{Pois}}(i > 0, k_{AB}\tau) = k_{AB} \exp(-k_{AB}\tau) \tag{2.14}$$

Then, our goal is to sample from such Poisson distributions, in order to generate representative samples of τ, which represents the time needed to observe a transition. To do so, we can use the inverse transform method. We start by calculating the cumulative distribution function of our Poisson distribution:

$$F^{\text{Pois}}(i > 0, k_{AB}\tau) = \int_{\tau}^{\infty} k_{AB} \exp(-k_{AB}\, \mathrm{d}t)\, \mathrm{d}t = \exp(-k_{AB}\tau) \tag{2.15}$$

Because time is continuous, such a cumulative distribution follows a uniform distribution $F^{\text{Pois}}(i > 0, k_{AB}\Delta t) \sim \mathcal{U}(0,1)$. Hence, to sample from $P^{\text{Pois}}(i > 0, k_{AB}\tau)$, it suffices to draw a uniform random number $u \sim \mathcal{U}(0,1)$ and solve for τ:

$$\tau = -\frac{\ln(u)}{k_{AB}} \tag{2.16}$$

It is straightforward to generalize this argument to a generic transition from a metastate i to any other accessible metastate j: the cumulative rate is $K_i = \sum_{i \neq j} k_{ij}$, and the time increment for a generic step of the Kinetic Monte Carlo algorithm starting at configuration i reads:

$$\Delta t = -\frac{\ln(u)}{K_i} \tag{2.17}$$

Note that the time increment does not depend on which process was sampled, but only on the cumulative rate of the processes available at a given step. Sampling the process for the algorithm to execute is actually simpler, and requires a second uniform random number $u' \sim \mathcal{U}(0,1)$, which is then normalized by the cumulative rate constant. For a generic step starting at a configuration i, one picks the process p such that:

$$\sum_{j=1}^{p-1} k_{ij} \leq u' K_i = u' \sum_{i \neq j} k_{ij} \leq \sum_{j=1}^{p} k_{ij} \tag{2.18}$$

This ensures that processes are chosen with the correct probability: faster processes — with a large rate constant — are sampled with a higher probability than slower ones — with a smaller rate constant. By doing so, one ensures that the generated trajectories satisfy the master equation 2.6.

In figure 2-6, we illustrate the schematic flowchart of a full, generic KMC algorithm. A more detailed representation of our implementation is provided in figure 3-9.

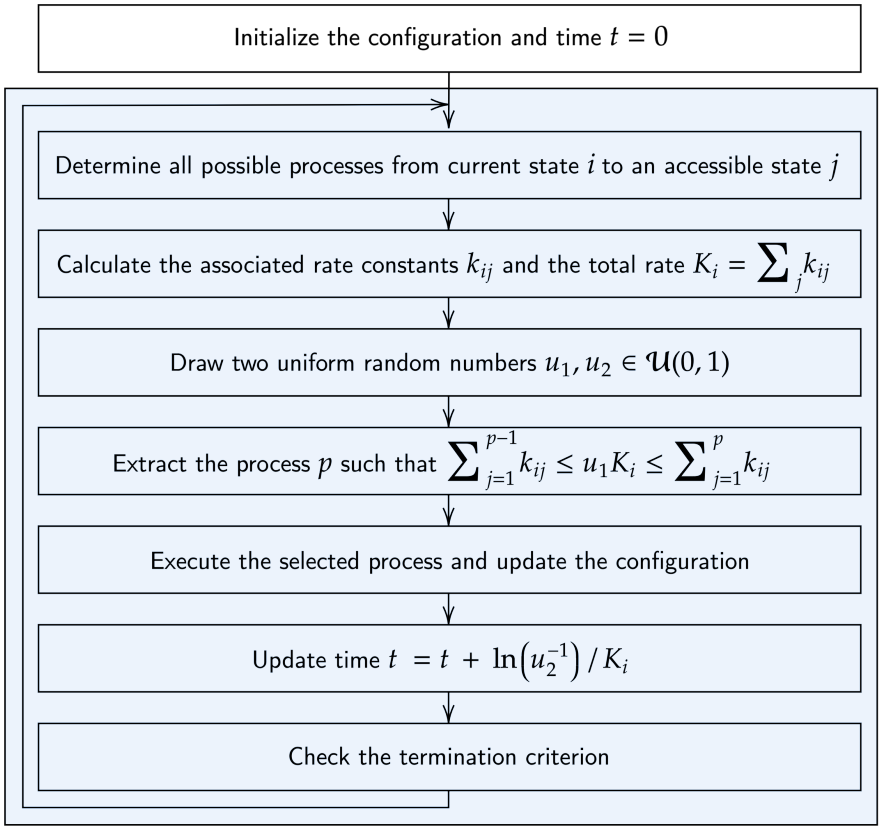

Figure 2-6: Schematic flowchart of a Kinetic Monte Carlo algorithm.

Chapter 3

Methodology

3.1 Off-Lattice Kinetic Monte Carlo

The goal of this chapter is to describe the methodology of our Kinetic Monte Carlo approach to the oxidation of hafnium. We start by justifying an important modeling choice that we have briefly discussed in section 2.3.1: instead of mapping the problem to a lattice, the Kinetic Monte Carlo algorithm used here operates off-lattice.

Mapping the structure to a lattice [MHS05] simplifies the algorithm drastically: by reducing memory (and thus also computational time) requirement, by providing a direct way to characterize neighborhoods of arbitrary size (and thus of enumerating processes), and by enabling reuse of energy barriers (and thus of rate constant) calculations. Indeed, because the processes are mapped to specific lattice configurations, the number of different transitions is finite, and all transitions can in principle be calculated and characterized in advance.

3.1.1 Motivation

The choice to develop an off-lattice KMC simulation was motivated by the following observations. First — and most importantly — hafnium dioxide (HfO_2) is typically amorphous at the high-temperature and pressure conditions that we are interested in. In other words, hafnium dioxide does not have the regular crystalline structure

needed for an on-lattice KMC simulation. The lack of long-range order in amorphous materials itself results in several unique physical and chemical properties. For example, amorphous materials typically have a higher degree of structural disorder, lower density, and higher specific surface area than their crystalline counterparts [Gas83]. Unfortunately, for the purposes of our KMC simulation, the mapping of such a problem involving an amorphous structure on a lattice pattern can generate several significant inaccuracies, stemming from an incorrect atomic configuration that results in erroneous representation of energies and forces.

Second, hafnium expands as it oxidizes. As highlighted by early-time molecular dynamics calculation, the reaction of hafnium with oxygen results in a volume increase in the material, leading to expansion. Such an expansion is of concern in materials science and engineering, as it can cause stress buildup and potentially lead to cracking or other types of structural damage [WCC+20]. From the KMC perspective, this leads to additional complications when trying to map even just the non-oxidized layers to a lattice: conserving and enforcing a pre-defined lattice structure to model a material that is expanding leads to inaccuracies in the description of atomic positions and in the generation of stress and strain. In other words, the calculated barriers will not be accurate if we do not allow for relaxation, and consequent modeling will bias the system's dynamics.

Third, a lattice model does not appropriately capture the evolution of the surface as the material oxidizes. The displacement and variation of adsorption sites is often difficult to track and predict, which makes it problematic even for off-lattice models, but on-lattice models lack the flexibility to address the complicated dynamics of an oxidizing surface. Off-lattice models allow more versatility in the treatment of oxide layers at the surface, in modeling the complex behaviours of surface re-arrangement, and in addressing the emergence of repulsive forces at the surface.

As a summary, an on-lattice Kinetic Monte Carlo framework is not well suited to our problem of interest, at least as long as we are interested in replicating the exact dynamical evolution of the system (as, for example, given by molecular dynamics simulations–see discussion in section 4.3). In a more coarse-grained model [VMG02]

in which accurate atomic positions are not so important, an on-lattice framework (or adaptive lattice) might offer a satisfactory compromise between accuracy and computational efficiency [HJ01]. For our purposes, the above-described limitations motivated the development of the off-lattice framework described below.

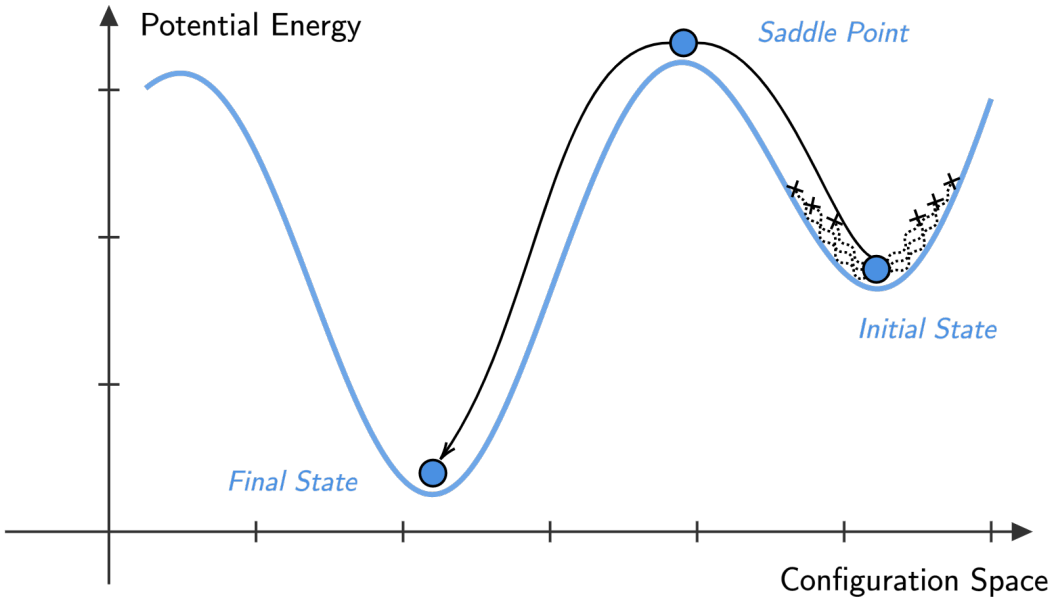

Figure 3-1: Schematic representation of an atom overcoming an energy barrier on a potential energy surface and transitioning from one state to another state.

3.1.2 Voronoi Tessellation

As we have underlined in section 2.3.1, one of the key challenges associated with a Kinetic Monte Carlo algorithm is the identification of neighborhoods. Consider for instance an atom located in a local energetic minimum during the material oxidation. Eventually, after a large number of escape attempts driven by thermal fluctuations, it will exit its potential well, traverse the transition state (on a N-th dimensional surface, the location of the transition state is also referred to as the saddle point), and reach another potential well beyond that energy barrier. MD simulations allow for a precise reconstruction of the atomic trajectory. KMC simulations, however, need

to characterize such translations not only by evaluating and weighing the energy barriers, but also by enumerating the total number of processes available, to make sure that the sampled dynamics are statistically accurate.

To discretize the number of possible events at the atomistic scale, we endow our Kinetic Monte Carlo framework with a Voronoi tessellation [Aur91, For95]. From a mathematical perspective, a Voronoi tessellation is a decomposition of a given space into a set of non-overlapping, convex regions based on the positions of a specified set of points. Formally, let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points in a d-dimensional Euclidean space $\mathbb{K}$. For all i, the Voronoi cell V_i of point p_i is the set of all points in the space $\mathbb{K}$ that are closer to p_i than to any other point in P:

$$V_i = \{x \in \mathbb{K} \mid \text{dist}(x, p_i) \leq \text{dist}(x, p_j),\ \forall j \neq i\} \tag{3.1}$$

where $\text{dist}(\cdot, \cdot)$ is the Euclidean distance over $\mathbb{K}$. Collectively, the Voronoi tessellation of P is the collection of the Voronoi cells of every point in P:

$$V_P = \{V_1, V_2, \ldots, V_n\} \implies \mathbb{K} = \bigcup_{i=1}^{n} V_i \tag{3.2}$$

Each Voronoi cell V_i is a convex polyhedron, and the Voronoi tessellation V_P divides the space $\mathbb{K}$ into n disjoint regions, one for each point in P. Mathematically, the vertices $\mathcal{V}(V_P)$ of such a tessellation are defined as the intersection points of the hyperplanes that define the boundaries of the Voronoi cells. Each vertex corresponds to a pair of neighboring points in the original point set P, and it is equidistant from those two points.

The relevance of such tessellations to atomistic modeling has been known for some time [WS33]. One proceeds by defining P to be the $\mathbb{R}^{3\times N}$ matrix containing the positions of the N atoms of a given configuration. The resulting Voronoi decomposition of P divides the space in N regions, and the vertices $\mathcal{V}(V_P)$ of the tessellation represent the points in space that are equidistant to the nearest neighboring atoms.

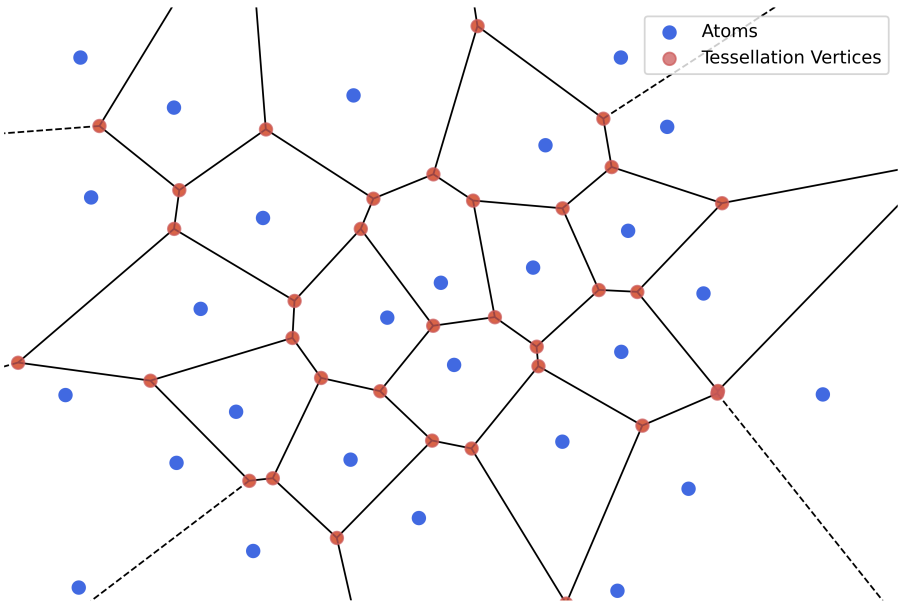

Figure 3-2: Two-dimensional Voronoi tessellation of an arbitrary set of atoms.

The tessellation is used as follows: at each step of our KMC algorithm, we generate a Voronoi tessellation based on the positions of the hafnium atoms. Then, the vertices of the tessellation represent the predicted stable positions for oxygen atoms (see figure 3-2). These predictions can be post-processed to improve algorithmic stability: for example, we can merge two tessellation points which are close to each other, and discard tessellation points which are too close to hafnium atoms (much closer than the bonding length, for instance, which would lead to strong repulsive forces). Once we obtain a simplified tessellation, the task of enumerating processes and keeping track of neighborhoods is greatly simplified, and in some sense, become very similar to an on-lattice framework: for every oxygen atom in our configuration, we can collect the predicted stable states within a given cutoff range, discard those which are already occupied by another oxygen atom, and we obtain the list of all possible destinations sites for our atom to translate. This results in a sort of discretization of the space which is similar to a dynamic lattice mapping, but nevertheless much more flexible and versatile, as it is recomputed at each step and does not include any positional

constraint, meaning that the experimental bond lengths are not enforced explicitly.

Such a procedure features obvious limitations: the predicted stable states are calculated in a purely geometrical fashion, without any energetic argument involved. This is why our KMC implementation includes a regular minimization of the configuration. Minimization refers to the process of finding the lowest energy configuration of a system of atoms or molecules by adjusting iteratively the positions of the atoms to minimize the total potential energy of the system, and thus converge towards the most stable configuration of the system. The most commonly used method for minimizing the potential energy of a system is the gradient descent method [JMJ98]. This involves calculating the gradient of the potential energy with respect to the atomic positions and iteratively moving the atoms in the direction of the negative gradient — and thus downhill on the potential energy surface — until the energy is minimized. This ensures that the predicted tessellation locations towards which we displace oxygen atoms are actually stable, by correcting angles, distances and separations between all atoms in the configuration.

On the other hand, the tessellation procedure is particularly cheap from a computational standpoint: the cost of computing the Voronoi tessellation of an atomic structure using the Qhull algorithm scales as $\mathcal{O}(N \log N)$, where N is the number of atoms [BDH96].

3.2 Dynamic Calculations of Rate Constants

By setting up the off-lattice framework of our algorithm, generating the tessellation and collecting the possible and available processes, we have addressed the first main challenge of a Kinetic Monte Carlo simulation, as described in section 2.3. The two other challenges — weighing the relative probability of each process and inferring time — depend directly on rate constants which we discuss extensively below.

In our case study of the oxidation of hafnium, as confirmed by molecular dynamics simulations and experiments, the dynamics of the oxidation can be decomposed into two main classes of processes: adsorption and diffusion. We start with a pristine

hafnium structure, which is progressively oxidized. As layers of hafnium dioxide start to form, the oxygen atoms start diffusing in the material. Diffusion is a much slower process compared to adsorption: as mentioned in section 2.3.3, adsorption is generally a non-activated process, in which particles impinging the surface have a sufficiently high velocity to surmount energy barriers. In contrast, diffusion is a slow, collective process in which individual atoms have to cross relatively high energy barriers to diffuse inside the material. Due to the differences between these processes, they require different treatments, as discussed below.

3.2.1 Adsorption Model

The Transition State Theory background justifying the treatment of adsorption as a non-activated process as well as the characterization of the related rate constant 2.10 was introduced in section 2.3.2.

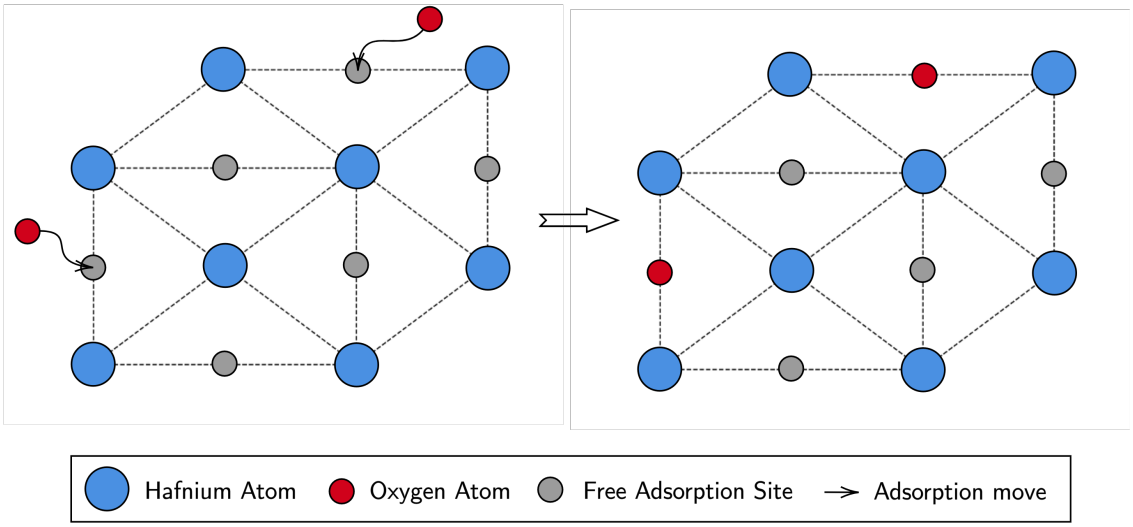

Figure 3-3: Two-dimensional representation of the simplified off-lattice adsorption model.

The adsorption model of our KMC implementation is therefore simple and schematic, and it is designed to capture the core physical aspect of the problem. At each step, as

we recalculate the Voronoi tessellation of the present configuration, the tessellation points which lie on the surface of the structure are considered as possible adsorption sites. To each adsorption site which is not already occupied by an oxygen atom, we assign a potential adsorption process. Collecting all possible processes, we obtain the list of possible adsorption moves which are then considered in the sampling and inference sections of the KMC algorithm.

This means that at the very early stages of the KMC simulation, the great majority of processes will be adsorption processes. As the surface becomes covered and the number of available diffusion moves increases (see section 3.2.2), the fraction of adsorption processes decreases.

For early times — that is, for times also accessed by our molecular dynamics simulations — we treat the sticking coefficient $S_{i,f}(T)$ of 2.10 as a constant: if a site is free and the process is sampled, an oxygen atom is adsorbed. However, we have noticed that this simplistic approximation introduces inaccuracies as the surface becomes more and more oxidized and transitions from a regular crystal structure to an amorphous form, most likely due to the emergence of repulsive forces originating from atoms at the surface (and the consequent transition towards the activation of the adsorption process).

As a first approximation for early times, we have not incorporated any correction for these effects. In other words, our adsorption model is given by:

$$k_{i,f}^{\text{ads}}(T, p_i) = \frac{p_i A_{\text{UC}}}{\sqrt{2\pi m_i k_B T}} \tag{3.3}$$

We intend to transition towards a more tailored approach in future work, in order to avoid biasing longer-times dynamics.

3.2.2 Atomistic Diffusion Model

As we have mentioned above, diffusion is a slower process: not only because it happens as a consequence of collective movements of atoms, but because such atoms need to overcome high energy barriers to translate. Therefore, in a KMC framework,

translation processes will advance the inferred simulation time significantly, much more than faster processes such as adsorption. Thus, it is particularly important to calculate transition rates accurately, and as we have detailed in section 2.3.2, the (exponentially) predominant quantity determining the rate constant of such activated processes is the energy barrier they need to overcome (see equation 2.8).

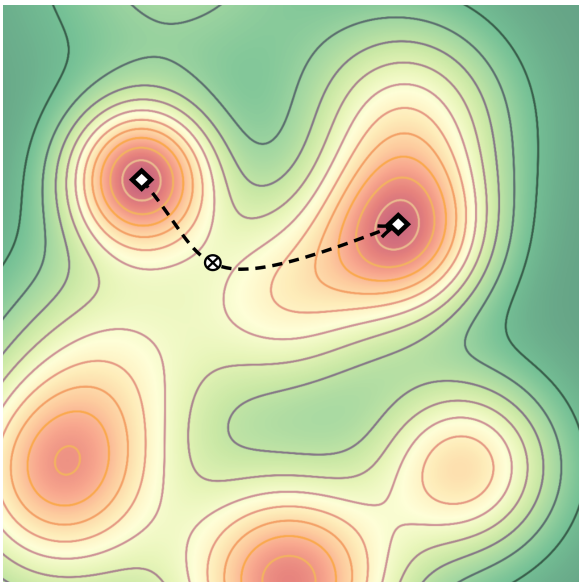

Figure 3-4: Schematic representation of an arbitrary two-dimensional potential energy surface with multiple energy minima (red wells), with initial and final states reproduced as diamonds and the saddle point of the (dashed) minimum energy path represented as a cross (adapted from [APR19]).

There are two classes of methods addressing the problem of determining energy barriers: interpolation methods and local methods [APR19]. When both the initial and the final state of a given elementary process are well-known, interpolation methods are generally preferable, and permit to analyze the potential energy surface between such states. When only the initial state is known, local methods are needed to take advantage of the local topology in order to explore the possible escape pathways from the initial state. Because our initial formulation did not include a description of the potential destination states given an initial state, our earlier efforts focused on local methods. By setting up a discretized off-lattice framework (as described in

section 3.1) which collects and describes all possible processes that can happen given an initial state, however, we could transition to an interpolation setting. Among interpolation methods, the two most popular types of methods are drag methods and chain-of-state methods [HJJ02].

Drag methods [HL77] are the simplest form of interpolation methods: the idea is to identify a small number of degrees of freedom — typically, in our case, the dimensions along which a given atom moves — which characterize the most significant differences between the initial and the final state, and then to optimize along those degrees of freedom while maintaining all others fixed. In considering a simple transition of a single atom from one metastate to another, one could consider as little as the three degrees of freedom describing the coordinate of the atom, for instance. Such methods are particularly successful in analyzing reaction paths dominated by one, two or three degrees of freedom, but are completely dependent on such choices of reaction coordinates, and tend to fail when applied to systems with multiple pathways or complex energy landscapes.

Chain-of-state techniques use strings of images, which are virtual replicas of the system in different positions (and thus configurations) of the reaction coordinate. The images are connected by springs, which represent the energy barriers that the system must overcome as it moves along the reaction coordinate. The free energy profile is then calculated by minimizing the energy of the string subject to the constraint that the images remain on the reaction coordinate (up to arbitrary thresholds for energies and forces). The most popular algorithm is the Nudged Elastic Band (NEB) method [JMJ98].

Formally, the chain of images is initiated as a chain $\mathcal{R}_{\text{NEB}} = [\mathbf{R}_0, \mathbf{R}_1, \ldots, \mathbf{R}_N]$ where the end points $\mathbf{R}_0$ and $\mathbf{R}_N$ are fixed and include the configurations corresponding to the initial and final energy minima — and thus represent the initial and final states. The goal of the NEB algorithm is to optimize the positions of the $N-1$ intermediate images along the minimum energy path to estimate the location of the saddle point. To do so, one minimizes the objective function of the elastic band [JMJ98], regrouping the energy of the intermediate images and featuring a penalty

term through a spring constant k:

$$G_{\mathrm{EB}}(\mathcal{R}_{\mathrm{NEB}}) = \sum_{i=1}^{N-1} E(\mathbf{R}_i) + \sum_{i=1}^{N} \frac{1}{2} k(\mathbf{R}_i - \mathbf{R}_{i-1})^2 \tag{3.4}$$

To decrease the sensitivity to the spring constant in the vicinity of the transition state, only the tangential component of the spring force is considered. We illustrate the NEB method on a sample PES in figure 3-5.

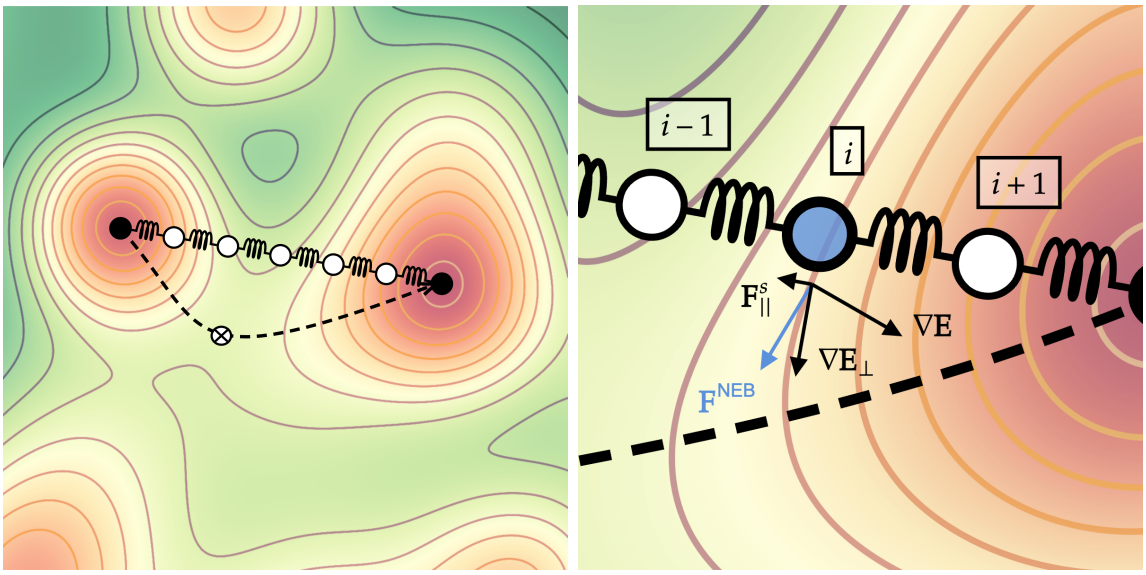

Figure 3-5: Illustration of the NEB method of a sample PES, with a closeup describing the composition of forces at a given step (adapted from [APR19]).

More specifically, the total force acting on a given image i at any given iterative step of the Nudged Elastic Band algorithm is:

$$\mathbf{F}_i^{\mathrm{NEB}} = \mathbf{F}_i^s|_{||} - \mathbf{F}_i|_{\perp} \tag{3.5}$$

where the true force acting on the image reads:

$$\mathbf{F}_i|_{\perp} = \nabla E(\mathbf{R}_i)|_{\perp} = \nabla E(\mathbf{R}_i) - \nabla E(\mathbf{R}_i) \cdot \hat{\boldsymbol{\tau}}_i \tag{3.6}$$

and the sum of the spring forces reads:

$$\mathbf{F}_i^s|_{||} = k\left(|\mathbf{R}_{i+1} - \mathbf{R}_i| - |\mathbf{R}_i - \mathbf{R}_{i-1}|\right)\hat{\boldsymbol{\tau}}_i \tag{3.7}$$

where $\hat{\boldsymbol{\tau}}_i$ is the tangent unit vector at image i.

Finally, an additional improvement comes from adding a Climbing-Image feature (CI-NEB) [HUJ00]. When using this feature, after a few iterations, one can identify the highest-energy image and drive it upward the potential energy surface — and thus toward the saddle point — by loosing spring forces and by reversing the component of the true force parallel to the chain. For such an image $i_{\max}$, the force reads:

$$\mathbf{F}_{i_{\max}}^{\text{NEB}} = -\mathbf{F}_{i_{\max}}|_{\perp} + \mathbf{F}_{i_{\max}}|_{||} = -\mathbf{F}_{i_{\max}} + 2\mathbf{F}_{i_{\max}}|_{||} \tag{3.8}$$

By doing so, the maximum energy image is not affected by spring forces and is free to move up the potential energy surface along the elastic band and down the potential energy surface perpendicular to the band, making the CI-NEB more robust on complex energy landscapes.

The structure of the NEB method makes it particularly easy to parallelize: one can distribute every image on a different processing core, requiring only modestly fast communication. On the other hand, a clear limitation of the NEB algorithm is the high computational cost: the procedure involves a large number of energy and force evaluations, the optimization of various configurations, and — typically — a large number of iterations.

To reduce the number of iterations, we propose in this work the addition of an initial pre-processing of the configuration, as we now describe. The main idea is to use a discretized approximation of the PES to provide a good initial guess for the NEB procedure.

Given an initial setting (so an initial state and a final state), we start by interpolating between the two states and building a matrix containing the values of energies at each point of the resulting three-dimensional grid. We illustrate this schematically in figure 3-6 for a sample two-dimensional PES, where the matrix entries are the dis-

cretization points (the matrix is two-dimensional for clarity; in our implementation, it is three-dimensional).

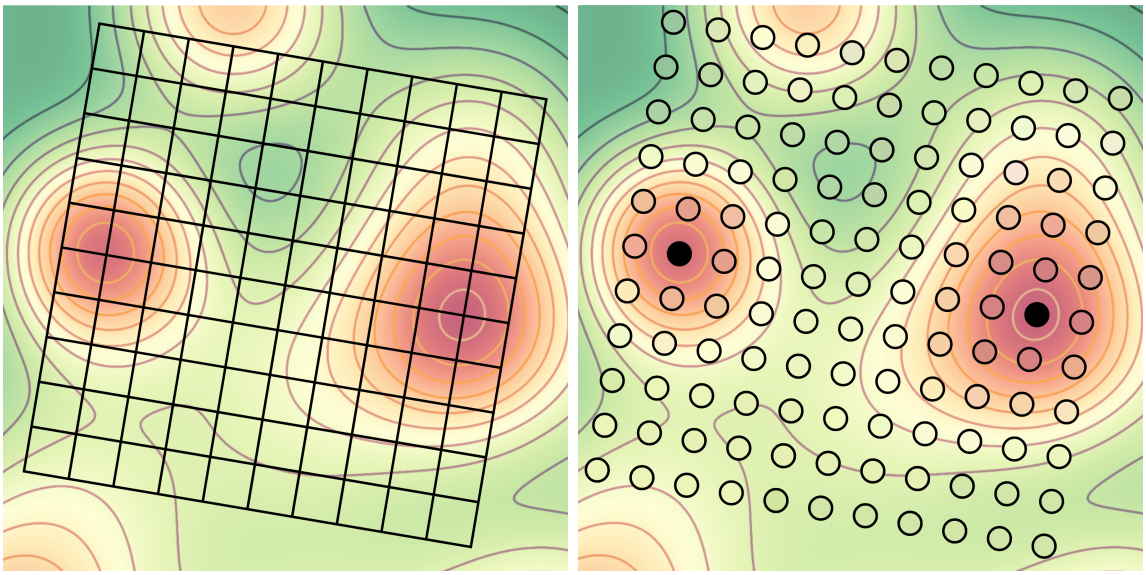

Figure 3-6: Discretization of the pointwise energy on a sample two-dimensional PES.

Then, we find an optimal path connecting the two states on the resulting high-dimensional grid, as shown in figure 3-7. Finally, we map the solution of the optimization problem back to the atomistic configuration: we use the discretized locations on the grid as initial locations for the replicas (see figure 3-8). In other words, we use the optimal path as an initial guess for the NEB calculation, instead of starting from a simple linear interpolation.

By doing so, we observe empirically that the number of iterations decreases drastically, because the initial guess is typically already close to the minimum energy path when starting the NEB calculation. Naturally, one needs to find a compromise regarding the number of discretization points: the higher the number of points, the higher the accuracy of the predicted initial guess, but also the higher the computational cost (due to the number of energy calculations and to the increase of images in the NEB calculation).

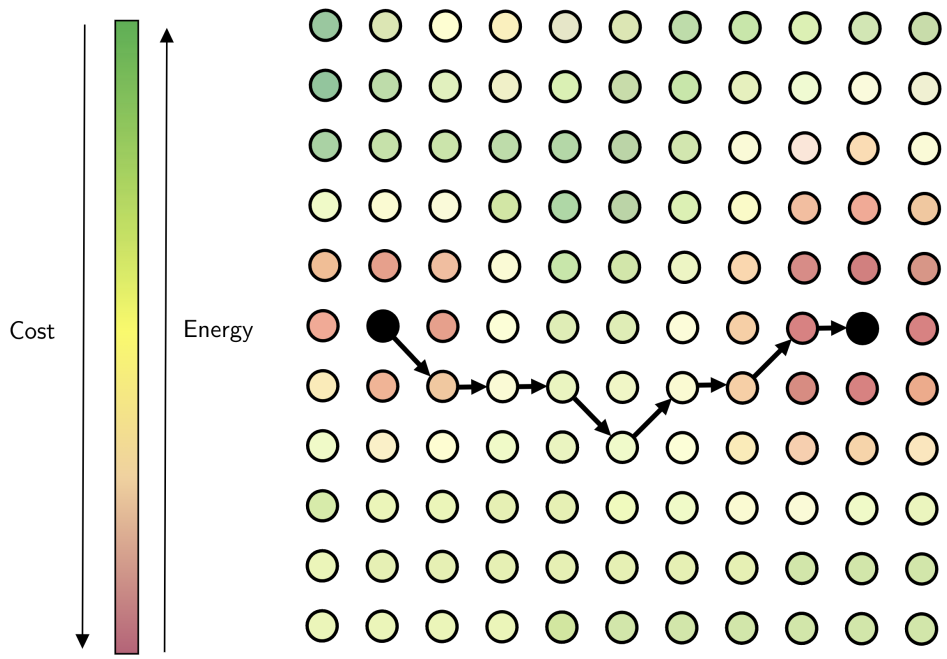

Figure 3-7: The discretized version of the atomistic problem can be transformed into an optimization problem: in the discretized framework, an appropriate initial guess for the NEB calculation is the optimal energetic path between the initial state and the final state. The grid is two-dimensional for clarity: in the atomistic framework, it is three-dimensional.

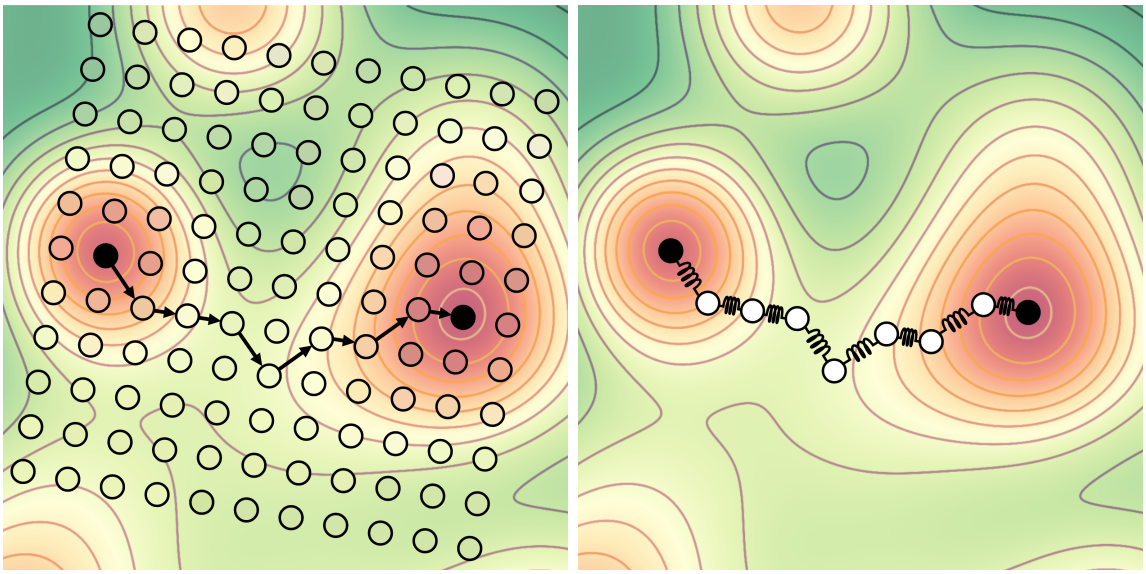

Figure 3-8: Finding the optimal path and using it as an initial guess for the NEB calculation.

With this integrated framework, we could compute energy barriers efficiently and thus to calculate dynamically the rate constants of the simplified Eyring equation:

$$k_{ij} = \frac{k_B T}{h} \exp\left(-\frac{E_{\mathrm{TS}_j} - E_i}{k_B T}\right) \tag{3.9}$$

where E_{TS_j} is the energy at the transition state between i and j, and E_i is the energy of the initial state.

3.2.3 Rejection Scheme

There are two fundamentally similar classes of Kinetic Monte Carlo algorithms, which differ in the way one treats the execution of processes and the calculation of rate constants at each move: the so-called rejection-KMC algorithms (rKMC) and rejection-free-KMC algorithms (rfKMC) [Jan12].

The standard version of the algorithm is rejection-free [Gil76] KMC: one calculates the rate constant of every possible process, and then choses the one to be executed according to the framework described in section 2.3.3. The alternative version includes a rejection scheme: at every move, only the rate constant of a single, randomly sampled process — among all possible processes — is calculated, and then the move is either accepted or rejected according to the relative probability of the sampled event.

To account for the bias in the uniform sampling of an event, one has to include in the acceptance step a judiciously chosen upper bound for the rate constant of the process; at an arbitrary step where the system lies at state i, we sample uniformly the event corresponding to a transition to a state j, we calculate the rate constant k_{ij}, and then we accept the process with probability [Jan12]:

$$\mathbb{P}_{ij}[\text{"accept the process"}] = \frac{k_{ij}}{k_i^0} \tag{3.10}$$

where k_i^0 is an upper bound for the transition from state i to any generic state i'.

In our implementation, when we ran simulations with such a rejection scheme in which we estimated the upper bound empirically — by running the rejection-free version for long enough — and in an adaptive fashion — by adapting the upper bound to the stage of the simulation. It is important to adapt the upper bound throughout the simulation, especially if the different classes of processes treated in a same KMC simulation (like adsorption and diffusion processes in our case study) feature important disparities in terms of rate constants: in specific regimes of the simulation where only fast processes can happen, it would be extremely inefficient to use the same upper bound as the one needed to address regimes in which slower processes can happen too. But despite such adaptations, one still needs to guarantee that k_i^0 is actually an upper bound, to avoid biasing the dynamics [Jan12].

A correction is also required for the time inference step. Suppose that we enumerate K_i possible different processes at stage i; then, we advance the simulation time by

$$\Delta t^{\text{rej}} = -\frac{\ln(u)}{K_i \cdot k_i^0} \tag{3.11}$$

where $u \sim \mathcal{U}(0,1)$. Crucially, we advance time in any case, whether the process was accepted or not. Also, it is important to note that the time increment Δt^{rej} is potentially much smaller than the time increment Δt of rejection-free algorithms described by equation 2.17.

It is quite straightforward to show that both methods converge to the same distributions and are statistically equivalent [Ser11, Sad84]. Rejection-free-KMC schemes may require more simulation time to converge because they involve the calculation of potentially large lists of possible processes. In contrast, rejection-KMC avoids such a computational burden, but advances the simulation clock by much smaller increments. What makes one version of the algorithm preferable to the other is essentially dictated by two variables: the time it takes to calculate rate constants (which itself depends on the number, variety and complexity of processes) and on how efficient the acceptance-rejection criterion can be designed (which depends on the variance of rate constants between processes). For our purposes, the fast changes in the orders

of magnitude of the rate constants — as part of the transition from an adsorption-controlled regime to a diffusion-controlled one — made it challenging to design an efficient rejection scheme. As a result, most of the calculations presented in chapter 4 were performed with a rejection-free procedure.

3.3 Implementation

In this section, we present an overview of the off-lattice Kinetic Monte Carlo framework developed here for simulating the oxidation of hafnium. A schematic flowchart of the implementation is shown in figure 3-9.

The code is written in Julia, as part of the CESMIX efforts to design an integrated computational framework to advance the state-of-the-art in predictive simulation [Git]. It is designed to be modular, highly parallelizable, and scalable.

The modularity feature appears in various forms: first and foremost, in terms of compatibility with — and integration of — various potentials. As discussed in section 2.1, the results obtained in section 4.2 are calculated with the COMB potential, mainly for validation purposes with our molecular dynamics simulations. Throughout our work, however, we have transitioned towards fast, tailored machine-learned potentials. Most notably, we integrated the possibility to run KMC simulations with a trained Proper Orthogonal Descriptors (POD) potential [NR23] on CPUs and with a trained Neural Equivariant Interatomic Potentials (NequIP/Allegro) potential [MBJ$^+$23] on GPUs. Due to the current cataloguing and characterization of processes, the code is naturally designed for simulation of oxidation, but it would be straightforward to implement new classes of processes: as long as one can describe their activated (or non-activated) character and an expression for the rate constant, any rare event at the atomistic level can be included. Input parameters include the simulation conditions (pressure and temperature), the description of the structure to analyze (the material, the size and experimental values of bond lengths to serve as initial guesses for the tessellation [BT15, GGS$^+$17, VSA$^+$14]), as well as simulation parameters (such as length of simulation and number of cores).

The resulting code is highly parallelizable, in the sense that it it decomposed in distinct collections of actions which can either run in parallel or cannot, as illustrated in figure 3-9. The actual atomic processes — the steps of the KMC algorithm — are executed sequentially: one process is treated at the time, and at most one atom is moved at every step. The treatment itself, however, is performed in parallel: the rate constants characterizing the available processes are calculated simultaneously through parallel Nudged Elastic Band calculations. Moreover, as discussed in section 4.2, NEB calculations themselves parallelize their images, making it a two-layer parallelism.

The parallelism associated with the NEB calculations is implemented through the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [TAB+22], while the parallelism associated to the distribution of the various NEB calculations across the cores is performed intrinsically in the algorithm. The structural updates, such as the structural relaxations and minimizations, is performed in LAMMPS in parallel as well, thanks to the very effective spatial weak-scaling of static atomistic simulation discussed in section 2.1.

Finally, the scalability feature comes in some sense as a consequence of the two previous arguments. In terms of spatial scalability, the code can be easily adapted, because most challenges introduced by a larger structure are already treated in a parallel fashion (minimization, calculation of energy barriers). Interestingly enough, however, the code is potentially scalable from a temporal perspective too: as will be discussed in greater detail in sections 4.2 and 4.3, the time-scale of interest in this study is dictated by validation purposes, and thus focused on shorter time scales. Therefore, processes that contribute relatively little to the time evolution of the system are considered and included thoroughly, in order to avoid biasing early-times dynamics. However, if one is interested in the simulation of events that are even less frequent, to access larger time scales, the general structure of the code (the off-lattice character, the rate constant treatment and the time inference) is still adapted, as one could alter the selection and filtering of atomic processes to focus on rarer events.

In terms of dependencies on external codes and libraries, the code is largely self-contained. The important exception is — as mentioned above — the use of LAMMPS

as an engine for energetic calculations and structure minimization. The minimization of the structure is actually performed through LAMMPS.jl, the Julia bindings to the LAMMPS API developed as part of CESMIX. The two other external libraries used in the code are the Julia bindings to the Atomic Simulation Environment (ASE), to generate the initial configuration, and the Julia bindings to the Qhull library, to generate the Voronoi decomposition of the space.

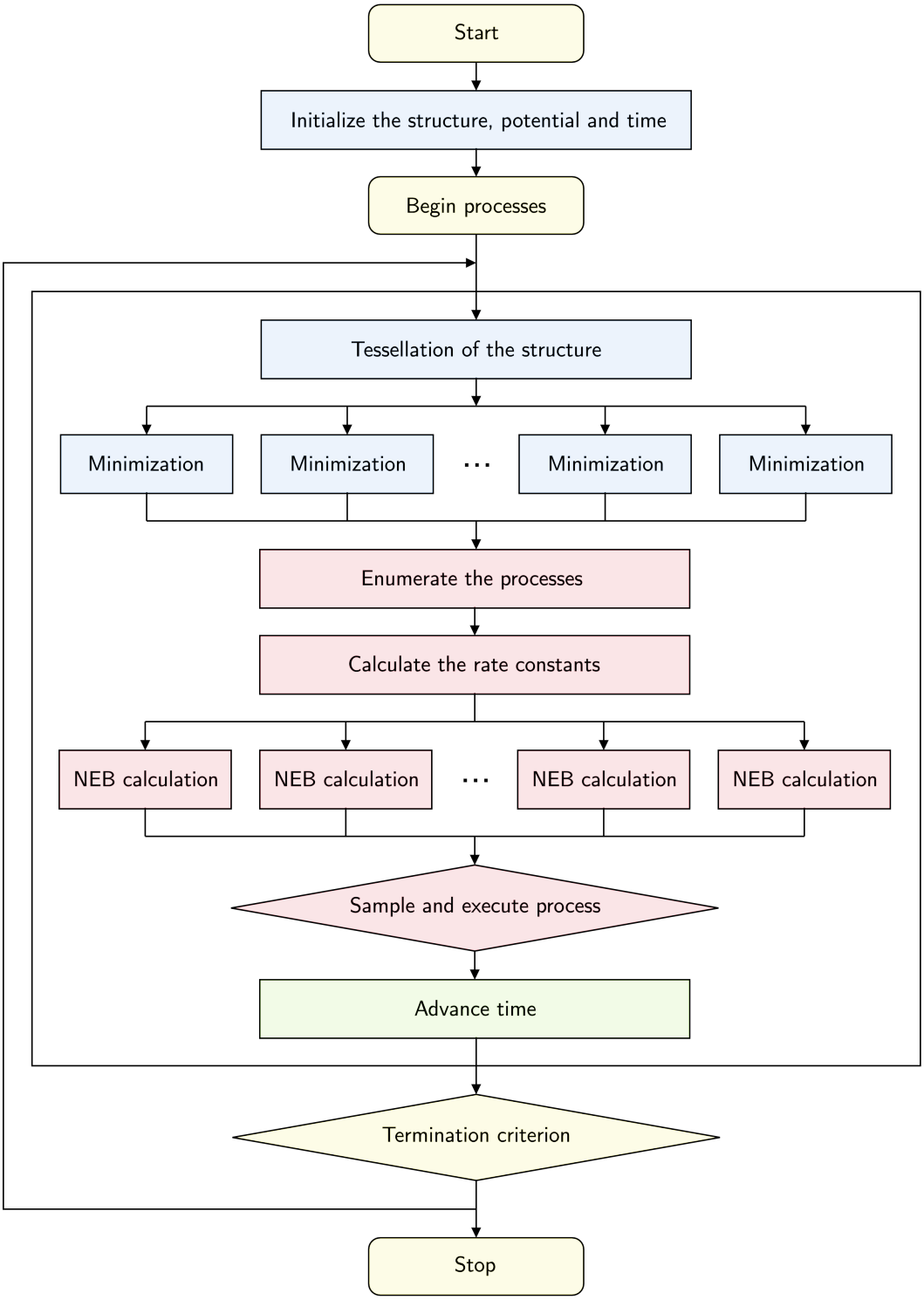

Figure 3-9: Flowchart of the off-lattice KMC implementation.

Chapter 4

Results

4.1 Molecular Dynamics Simulation of Oxidation

The goal of this chapter is to validate our KMC methodology and implementation by comparing its results for hafnium oxidation to MD simulations of the same system. We will start by describing the results of our Molecular Dynamics simulations.

To accelerate the dynamics and to approach the experimental conditions of interest, the longest MD runs were performed at $T = 2400$ K, $p = 1.013$ bar, just below the melting point of the material. The simulation setup is as follows: we built in LAMMPS a simulation cell containing a hafnium surface in contact with an oxygen gas phase. Hafnium has a Hexagonal Close-Packed (HCP) crystal structure at room temperature and ambient pressure. However, at such extreme conditions, hafnium undergoes a phase transition and transforms into a Body-Centered Cubic (BCC) crystal structure [HZZ$^+$11]. In the BCC crystal structure, hafnium atoms are arranged in a cubic lattice with one atom at each corner of the cube and one atom at the center of the cube. This results in a total of two atoms per unit cell. The BCC structure is characterized by a coordination number of 8, meaning that each hafnium atom is surrounded by eight nearest-neighbor atoms. The simulation cell is periodic in two dimensions (x and y). The oxygen gas phase is represented by individual oxygen molecules with initial velocity based on a Maxwell-Boltzmann velocity distribution.

At the bottom of the simulation cell, we include a fixed sheet of hafnium — to

stabilize the system — which is also thermostated — to keep temperature constant. All hafnium layers that are treated dynamically in the simulation rest on top of such layers.

As explained and justified in section 2.1, the interatomic potential used in this simulation is the Charge-Optimized Many-Body (COMB) potential.

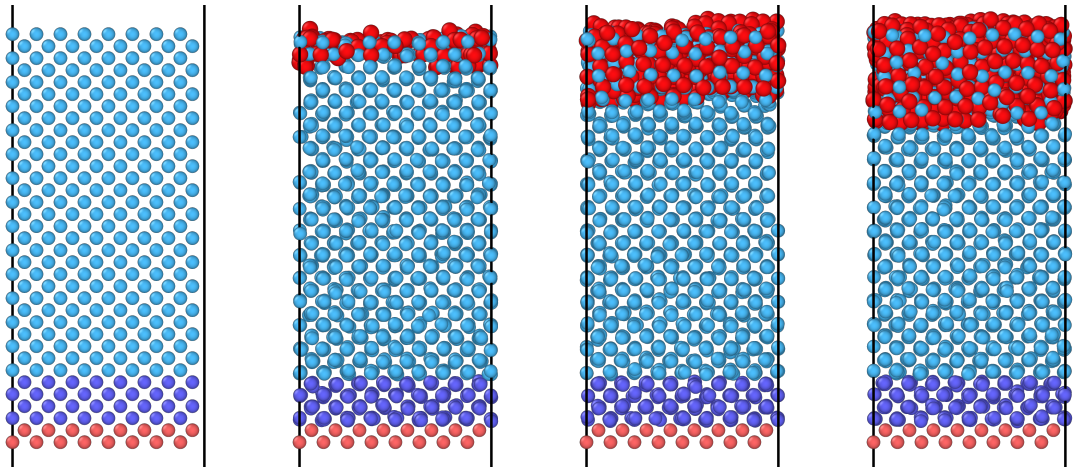

Figure 4-1: Snapshots of our MD simulation of the oxidation of hafnium at $T = 2400$K at various timesteps.

The simulation was performed on a relatively small system, running on a 32-core 3970x architecture. The KMC simulation will be performed on a system of the same size.

The simulation results were in line with our expectations: molecular dynamics simulations allow to replicate the kinetics with fidelity and provide key insights into the dynamics of the oxidation process. However — because they need to resolve atomic vibrations — they are very slow: our longest MD simulations ran for six months and managed to simulate 500 nanoseconds of system evolution.

Among the metrics that can describe significant collective dynamics of the oxidation process, the oxidation rate is particularly useful. In figure 4.1, we observe the characteristic oxidation behaviour that we expected: at early stages, the dynamics are dominated by the fast adsorption of oxygen on the initially pristine surface. As the

surface gets covered, an oxide layer forms, and slower processes — such as diffusion inside the structure — become more significant, and the dynamics become diffusion-controlled. The oxidation rate is expected to continue decreasing: as the thickness of the oxide layer increases, the probability for a vacancy to form at the surface gets smaller and smaller. Note that there is a direct parallelism between the diffusion of oxygen atoms downwards in the structure and the the diffusion of vacancies upwards towards the surface: the formation of vacancies at the surface is the consequence of a collection of individual oxygen translations.

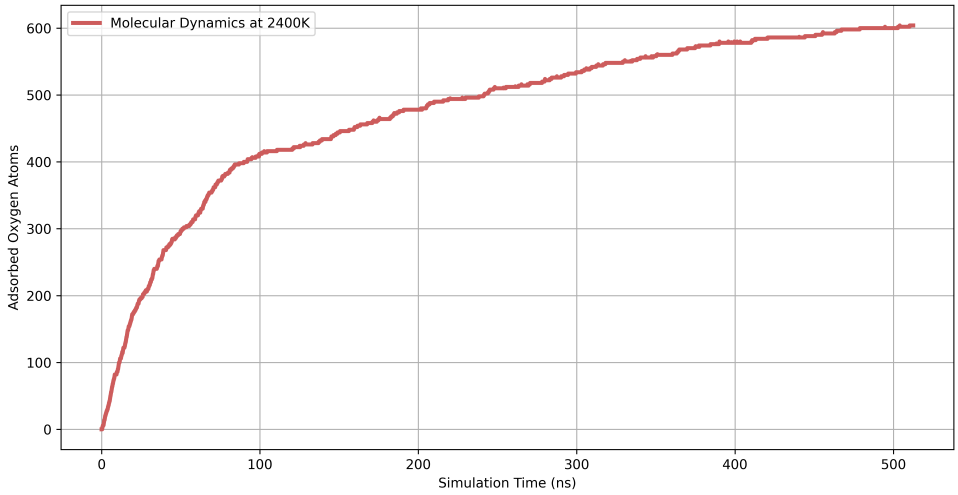

Figure 4-2: Adsorbed oxygen atoms as a function of the MD simulation time.

4.2 Kinetic Monte Carlo Simulation of Oxidation

Our Kinetic Monte Carlo simulations of the system were performed following the methodology described in chapter 3. Using the same 32-core 3970x architecture, the simulation of the first 500 nanoseconds of the oxidation process (of a domains of the same size and at the same conditions as the MD simulations) was performed in

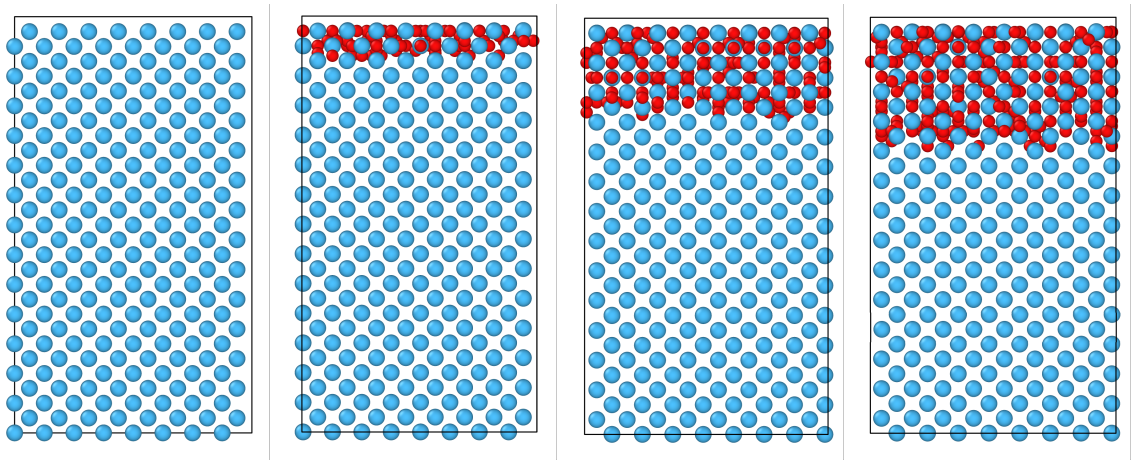

Figure 4-3: Snapshots of our KMC simulation of the oxidation of hafnium at $T = 2400$K at various timesteps (the same as figure 4-1).

approximately two hours, achieving a speed-up of approximately 2000 in terms of computational speed.

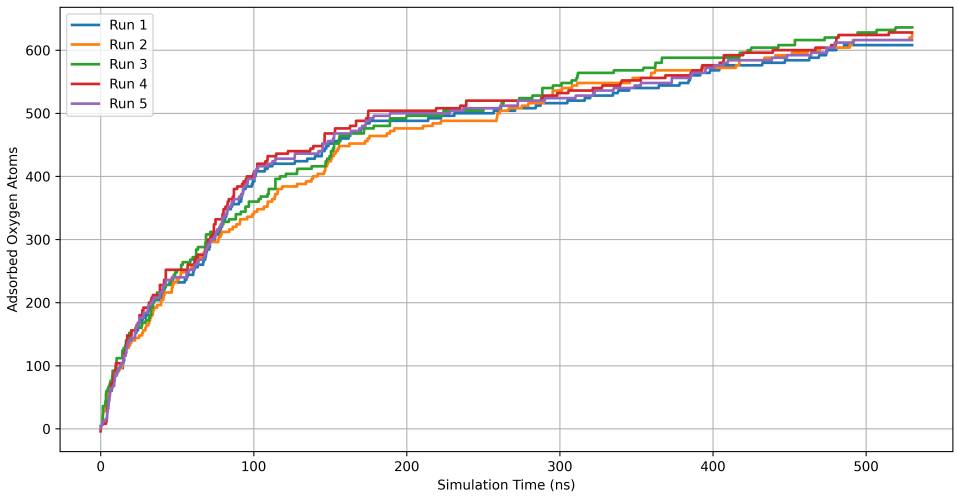

Figure 4-4: Adsorbed oxygen atoms as a function of the KMC simulation time for various distinct KMC runs with the same initial conditions.

Although detailed comparisons will follow in the validation section 4.3, here we note the existence of the two oxidation regimes (see figure 4-4): the oxidation is first adsorption-controlled, with high oxidation rates for early times, then transitions

towards a diffusion-controlled regime. Note that KMC has a stronger stochastic component than MD: it is expected to replicate the atomistic dynamics on average, but individual runs can naturally feature quite different trajectories.

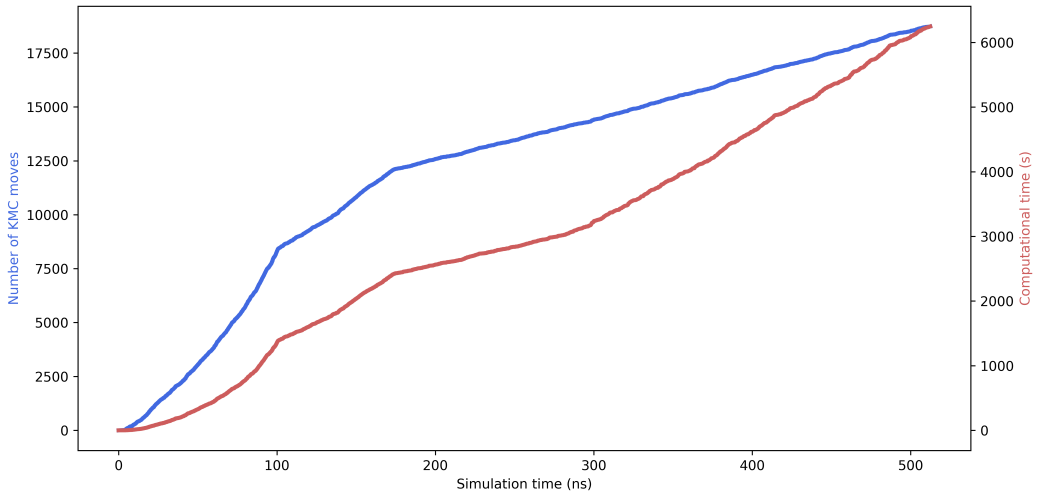

Figure 4-5: Number of KMC moves and empirical computational time as a function of the KMC simulation time.

Finally, we make a few remarks on the behavior of the KMC simulation in terms of computational time and simulation time. Consider figure 4-5. We are interested in the evolution of two measures throughout the oxidation simulation: the number of moves within the KMC algorithm and the computational time. The first quantity highlights — once again, quite clearly — the existence of different regimes in the oxidation dynamics: at early times, fast processes such as adsorption moves are sampled with much higher probability (because of the disparity in the rate constants and because at the beginning many adsorption sites are available and few diffusion moves possible). As the surface is progressively covered, fast processes become more rare, and thus single processes (such as high-barrier diffusion processes) advance the simulation time more significantly. Conversely, regarding the computational time, as the number of activated processes increases (each activated process requires the calculation of an

energy barrier through the adaptive Nudged Elastic Band procedure described in section 2.3.2), the time required to tackle a single process increases significantly. The growth is slower than one could expect because the increased amount of oxygen atoms adsorbed in the structure does not lead directly to an increase in the number of diffusion processes available: such a variable is dictated by the concentration of vacancies inside the structure.

4.3 Validation

A main objective of this research effort was validation of the Kinetic Monte Carlo results using Molecular Dynamics simulations. Such a goal is somewhat ambiguous, because MD serves both as a validation tool and as a training instrument, considering that the MD data is practically the only reliable atomistic description of the oxidation dynamics available.

The aim is to extract as many insights as possible from the molecular dynamics simulation to generate a coarse-grained framework that can not only replicate the early-times dynamics, but also simulate the oxidation behavior at longer times by perpetuating and extrapolating the same elementary processes. Hence, validation serves as a confirmation that the coarse-grained description does indeed replicate the early-times dynamics with fidelity.

As we have mentioned in section 4.2, one of the most instructive metrics of comparisons is that of oxidation rates. Figure 4-6 shows that the oxidation rates predicted by the two methods are very similar: despite small discrepancies in the adsorption-controlled regime, the KMC simulation reproduces well the transition to the diffusion-controlled regime, both in terms of timing and in terms of slopes.

Additional and independent validation can be obtained by comparing oxygen concentration profiles as a function of distance from the material surface. A comparison between the concentration profiles of oxygen in the structure as the MD and KMC simulations evolve is displayed in figure 4-7.

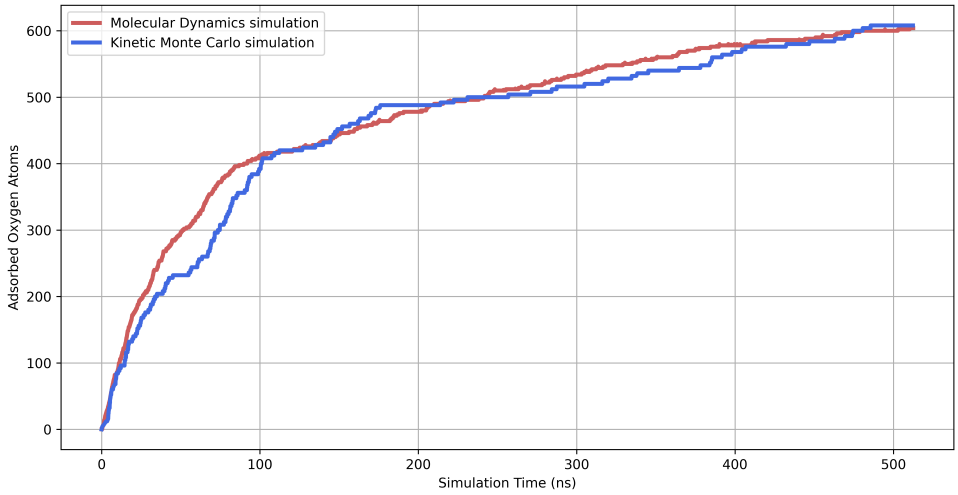

Figure 4-6: Comparison of the oxidation rates predicted by MD and KMC simulations at the same experimental conditions.

The figure shows that the KMC model is coarser in its treatment of successive oxide layers, most likely as a consequence of the soft constraints enforced by the tessellation grid. We observe for example a larger accumulation of oxygen atoms near the surface in the KMC model than in the MD model, and a more homogeneous distribution of the oxygen density in the oxide layer in the latter. Nevertheless, it is also worth noting that the speed of the diffusion — in terms of hafnium unit cells oxidized as a function of time — is very similar, resulting in a similar oxidation resistance.

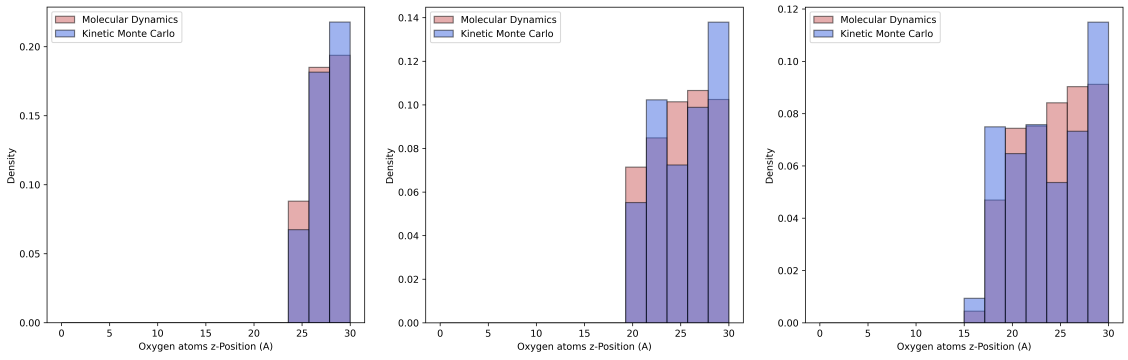

Figure 4-7: Comparison between the concentration profiles of oxygen in the structure as the MD and KMC simulations evolve (surface is at $z = 30$).

Finally, a third validation metric we can use is the analysis of Radial Distribution Functions (RDFs). Radial Distribution Functions describe the probability density of distances between particles in a system. In that sense, they provide important insights on the characterization of structural properties, by quantifying structural features such as coordination numbers, bond lengths and bond angles. Thus, by comparing the RDFs obtained from KMC simulations to the RDFs obtained with MD simulations, we can validate the accuracy of the KMC model in capturing the structural properties of the system.

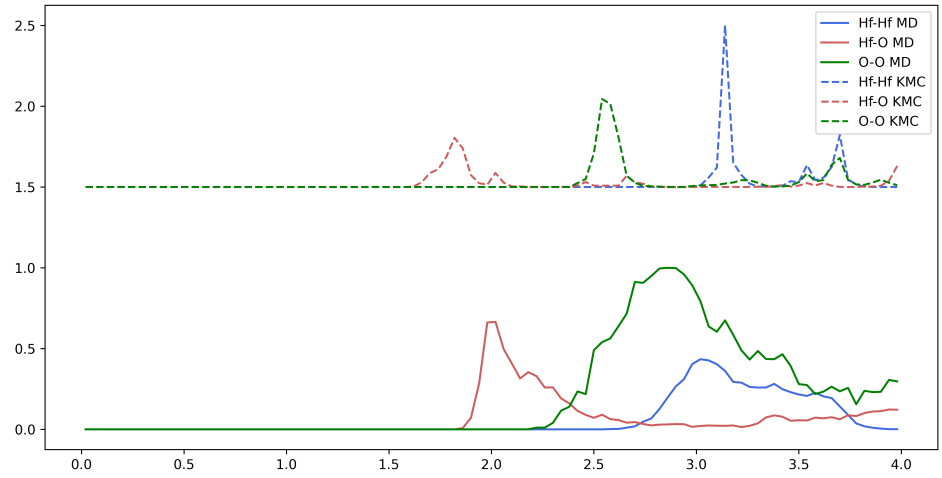

Figure 4-8: Comparison of the raw Radial Distributions Functions of MD and KMC simulations (KMC RDFs are shifted upwards for comparison purposes).

Figure 4-8 highlights that the RDFs produced by the two methods are not very similar. In particular, the RDF peaks — representing the most likely distance between two pairs of atoms — are relatively close, but the widths of the distributions are quite different: this is due to the fact that despite regular structural minimizations, the tessellation-based KMC imposes harder constraints on the atomic positions than one would expect. This is likely due to the intrinsically geometric nature of

the method: tessellation points will always be as far as possible from any other surrounding atom, before being driven downward the potential energy surface by the minimization process. The RDFs of the MD simulations look different: the peaks of the distribution match those of the KMC (highlighting the fact that the equilibrium bond lengths are close), but the widths of the curves differ significantly. This is most likely due to the constant atomic vibrations that atoms undergo in the MD algorithm, that end up increasing the variance around the equilibrium peaks.

Interestingly enough, by adding some noise to all atomic positions after tessellation calculations but before minimization the agreement between the two methods improves (see figure 4-9). More work is needed to understand the precise origin of the discrepancy between the KMC and MD results and how a KMC formulation which reproduces MD RDFs — assumed here to be the ground truth — can be developed.

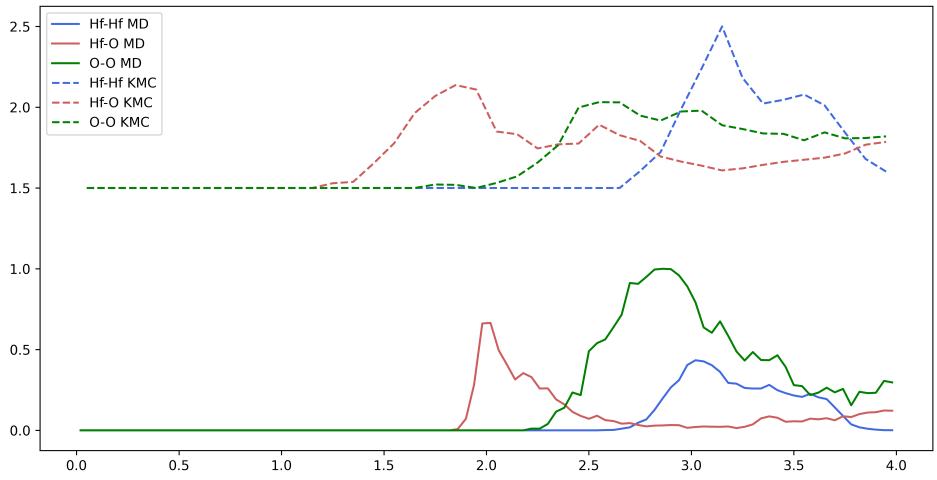

Figure 4-9: Comparison of the Radial Distributions Functions of MD and KMC simulations when allowing for artificial vibrations.

To assess the robustness of our KMC methodology, we have also performed simulations at a different temperatures, $T = 2000$ K. Crucially, the phase transition from the HCP crystal structure to the BCC crystal structure occurs around $T_{\mathrm{PT}} = 2030$

K [Jam63, KRES18]. Figure 4-10 is a comparison of the predicted oxidation rates at the two different temperatures, above and below the phase transition. In figure 4-11, we focus on the 2000 K run in particular and validate the oxidation rate predicted by MD and KMC simulations as well.

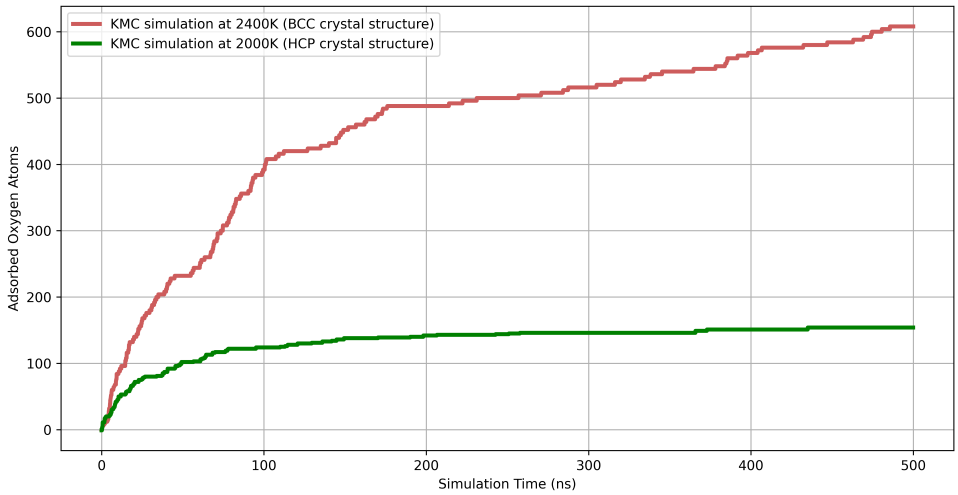

Figure 4-10: Comparison of the oxidation rates predicted by KMC runs at different temperatures.

We can see that the two methods predict very similar oxidation rates for early times. This is a particularly important confirmation from an algorithmic standpoint: as we could observe in figure 4-10, the oxidation rates differ significantly at different temperatures, mainly because of the different crystal structures. To be able to match with relatively good accuracy the early-time dynamics in both cases without changing anything but the initial crystal structure is a demonstration of the robustness and of the versatility of our KMC framework.

Overall, the off-lattice Kinetic Monte Carlo scheme appears to reproduce the early-time dynamics with fidelity, especially from a mesoscopic perspective. At the microscopic scale, our Kinetic Monte Carlo simulations are replicating with a fair accuracy

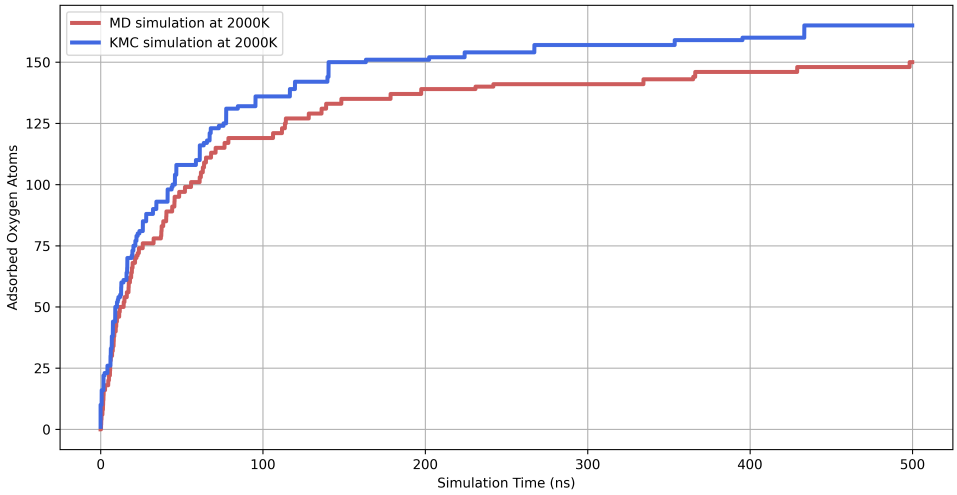

Figure 4-11: Comparison of the oxidation rates predicted by MD and KMC simulations at $T = 2000$ K.

the expected structural properties (the benchmark and validation instrument being the Molecular Dynamics simulation), such as radial distribution functions, bond lengths, and coordination numbers. More importantly, from the long-time integration perspective, our Kinetic Monte Carlo simulations appear to reproduce with fidelity dynamic properties at the mesoscopic scale, such as oxidation rates. Although this specific study focuses on the early-times dynamics, the significant computational speed-up (approximately $\times 2000$ compared to Molecular Dynamics calculations) is promising for tackling longer-times dynamics.

Chapter 5

Conclusion

In conclusion, we have constructed an off-lattice Kinetic Monte Carlo framework to bridge the large range of timescales present when simulating oxidation of hafnium at the atomistic level.

Our approach features a dynamic tessellation to account for the expansion of the underlying structure, accelerated estimation of rate constants by optimizing Nudged Elastic Band calculations and a scalable and modular Julia implementation integrated within the CESMIX software framework.

The presented results are exclusively focused on the early-time dynamics and on the validation effort with respect to Molecular Dynamics calculations. Future work will focus on the following challenges.

First, long-time integration remains a challenge: the current model is fast, accurate and robust at early times, but the evolution of metrics such as computational time as a function of simulation time (see figure 4-5) suggests that such an approach — reproducing structural properties with fidelity and including and considering fast processes — might reach computational bottlenecks preventing it to access longer time-scales. In future work we plan to investigate whether formulations with a progressive increase in the level of coarse-graining are possible and provide a worthwhile performance improvement.

Second, validation at longer time scales is very challenging since those timescales cannot be accessed by our molecular dynamics simulations, considered here as ground

truth. The wide time-scale gap between molecular dynamics simulations (on the order of nanoseconds) and experimental results (usually on the order of seconds) can be potentially bridged by Kinetic Monte Carlo methods using successive coarse-graining, as discussed above, for example. Unfortunately, results in this intermediate regime would be hard to validate.

Finally — in parallel with our simulation efforts — our goal is to develop an uncertainty quantification framework coupled with the Kinetic Monte Carlo scheme, in order to provide intrinsic metrics to evaluate the fidelity of Kinetic Monte Carlo simulations. In such a multi-scale and hierarchical effort, it will be important to extend the reasoning at all levels: by quantifying the impact of inherited errors (from Density Functional Theory calculations and Inter-Atomic Potentials, for instance); by formalizing the approximations in the current Kinetic Monte Carlo mathematical model; and by quantifying the uncertainty associated with key steps of our Kinetic Monte Carlo framework (namely, the generation of the tessellation and associated geometric approximations, the criteria for the enumeration of processes, the number of saddle point searches for a given process and the uncertainty in the evaluation of energy barriers, as discussed in sections 2.3.1, 3.1 and 3.2).

Bibliography

[APR19] Mie Andersen, Chiara Panosetti, and Karsten Reuter. A Practical Guide to Surface Kinetic Monte Carlo Simulations. *Frontiers in Chemistry*, 7:202, April 2019.

[AT17] M. P. Allen and D. J. Tildesley. *Computer simulation of liquids*. Oxford University Press, Oxford, United Kingdom, second edition edition, 2017.

[Aur91] Franz Aurenhammer. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, September 1991.

[AY20] Wanda Andreoni and Sidney Yip, editors. *Handbook of Materials Modeling: Methods: Theory and Modeling*. Springer International Publishing, Cham, 2020.

[BDH96] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, December 1996.

[BH10] Kurt Binder and Dieter W. Heermann. *Monte Carlo Simulation in Statistical Physics: An Introduction*, volume 0 of *Graduate Texts in Physics*. Springer, Berlin, Heidelberg, 2010.

[BKC12] Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. 2012.

[BM96] Gert D. Billing and Kurt V. Mikkelsen. *Introduction to molecular dynamics and chemical kinetics*. Wiley, New York, 1996.

[BP07] Jörg Behler and Michele Parrinello. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters*, 98(14):146401, April 2007.

[BT15] Philippe Blaise and Boubacar Traore. Structural Properties and Thermodynamics of Hafnium sub-oxides in RRAM. 2015.

[CDS81] J. M. Combes, P. Duclos, and R. Seiler. The Born-Oppenheimer Approximation. In G. Velo and A. S. Wightman, editors, *Rigorous Atomic*

and Molecular Physics, NATO Advanced Study Institutes Series, pages 185–213. Springer US, Boston, MA, 1981.

[CN17] Ib Chorkendorff and Johannes W. Niemantsverdriet. *Concepts of modern catalysis and kinetics*. WILEY-VCH Verlag GmbH & Co KG, Weinheim, third, completely revised and enlarged edition edition, 2017.

[DB84] Murray S. Daw and M. I. Baskes. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Physical Review B*, 29(12):6443–6453, June 1984.

[DG12] Reiner M. Dreizler and Eberhard K. U. Gross. *Density Functional Theory: An Approach to the Quantum Many-Body Problem*. Springer Science & Business Media, December 2012. Google-Books-ID: t6PvCAAAQBAJ.

[ED11] Eberhard Engel and Reiner M. Dreizler. *Density Functional Theory: An Advanced Course*. Theoretical and Mathematical Physics. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[Eyr35] Henry Eyring. The Activated Complex in Chemical Reactions. *The Journal of Chemical Physics*, 3(2):107–115, February 1935.

[For95] Steven Fortune. Numerical Stability Of Algorithms For 2D Delaunay Triangulations. *International Journal of Computational Geometry & Applications*, 05(01n02):193–213, March 1995.

[Gas83] P. H. Gaskell. Models for the structure of amorphous metals. In Hans Beck and Hans-Joachim Güntherodt, editors, *Glassy Metal II: Atomic Structure and Dynamics, Electronic Structure, Magnetic Properties*, Topics in Applied Physics, pages 5–49. Springer, Berlin, Heidelberg, 1983.

[GGS+17] Leighanne Gallington, Yasaman Ghadar, Lawrie Skinner, J. Weber, Sergey Ushakov, Alexandra Navrotsky, Alvaro Vazquez-Mayagoitia, Joerg Neuefeind, Marius Stan, John Low, and Chris Benmore. The Structure of Liquid and Amorphous Hafnia. *Materials*, 10(11):1290, November 2017.

[Gil76] Daniel T Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, December 1976.

[Gil77] Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, December 1977.

[Gil07] Daniel T. Gillespie. Stochastic Simulation of Chemical Kinetics. *Annual Review of Physical Chemistry*, 58(1):35–55, May 2007.

[Git] CESMIX Github. cesmix-mit.

[HJ01] Graeme Henkelman and Hannes Jónsson. Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table. *The Journal of Chemical Physics*, 115(21):9657–9666, December 2001.

[HJJ02] Graeme Henkelman, Gísli Jóhannesson, and Hannes Jónsson. Methods for Finding Saddle Points and Minimum Energy Paths. In Steven D. Schwartz, editor, *Theoretical Methods in Condensed Phase Chemistry*, Progress in Theoretical Chemistry and Physics, pages 269–302. Springer Netherlands, Dordrecht, 2002.

[HJMM93] J. M. Haile, Ian Johnston, A. John Mallinckrodt, and Susan McKay. Molecular Dynamics Simulation: Elementary Methods. *Computers in Physics*, 7(6):625, 1993.

[HL77] Thomas A. Halgren and William N. Lipscomb. The synchronous-transit method for determining reaction pathways and locating molecular transition states. *Chemical Physics Letters*, 49(2):225–232, July 1977.

[HUJ00] Graeme Henkelman, Blas P. Uberuaga, and Hannes Jónsson. A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *The Journal of Chemical Physics*, 113(22):9901–9904, December 2000.

[HZZ+11] YanJun Hao, Jun Zhu, Lin Zhang, HaiSheng Ren, and JianYing Qu. Structure phase transition and elastic properties of hafnium: First-principles study. *Philosophical Magazine Letters*, 91(1):61–69, January 2011.

[Jam63] John C. Jamieson. Crystal Structures of Titanium, Zirconium, and Hafnium at High Pressures. *Science*, 140(3562):72–73, April 1963.

[Jan12] A. P. J. Jansen. Kinetic Monte Carlo Algorithms. In A.P.J. Jansen, editor, *An Introduction to Kinetic Monte Carlo Simulations of Surface Reactions*, Lecture Notes in Physics, pages 37–71. Springer, Berlin, Heidelberg, 2012.

[JMJ98] Hannes Jónsson, Greg Mills, and Karsten W. Jacobsen. Nudged elastic band method for finding minimum energy paths of transitions. In *Classical and Quantum Dynamics in Condensed Phase Simulations*, pages 385–404, Lerici, Villa Marigola, June 1998. World Scientific.

[Jon24] J. E. Jones. On the determination of molecular fields. —II. From the equation of state of a gas. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 106(738):463–477, October 1924.

[Kra09] Peter Kratzer. Monte Carlo and kinetic Monte Carlo methods, April 2009. arXiv:0904.2556 [cond-mat, physics:physics].

[KRES18] Yu. V. Khlebnikova, D. P. Rodionov, L. Yu. Egorova, and T. R. Suaridze. Crystallographic Features of the -Phase Structure in Hafnium and Hafnium–Titanium Alloys. *Technical Physics*, 63(12):1771–1783, December 2018.

[KSDK18] S. V. Kolesnikov, A. M. Saletsky, S. A. Dokukin, and A. L. Klavsyuk. Kinetic Monte Carlo Method: Mathematical Foundations and Applications for Physics of Low-Dimensional Nanostructures. *Mathematical Models and Computer Simulations*, 10(5):564–587, September 2018.

[Lai87] Keith James Laidler. *Chemical kinetics*. Harper & Row, New York, 3rd ed edition, 1987.

[Li05] Ju Li. Basic Molecular Dynamics. In Sidney Yip, editor, *Handbook of Materials Modeling*, pages 565–588. Springer Netherlands, Dordrecht, 2005.

[MBJ+23] Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J. Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14(1):579, February 2023.

[MHS05] D.R. Mason, T.S. Hudson, and A.P. Sutton. Fast recall of state-history in kinetic Monte Carlo simulations utilizing the Zobrist key. *Computer Physics Communications*, 165(1):37–48, January 2005.

[MS05] Georges Martin and Frédéric Soisson. *Kinetic Monte Carlo Method to Model Diffusion Controlled Phase Transformations in the Solid State*. January 2005. ADS Bibcode: 2005homm.book.2223M.

[MU49] Nicholas Metropolis and S. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, September 1949.

[NR23] Ngoc Cuong Nguyen and Andrew Rohskopf. Proper orthogonal descriptors for efficient and accurate interatomic potentials. *Journal of Computational Physics*, 480:112030, May 2023.

[PPV20] Steven J. Plimpton, Danny Perez, and Arthur F. Voter. Parallel algorithms for hyperdynamics and local hyperdynamics. *The Journal of Chemical Physics*, 153(5):054116, August 2020.

[Reu11] Karsten Reuter. First-Principles Kinetic Monte Carlo Simulations for Heterogeneous Catalysis: Concepts, Status, and Frontiers. In Olaf Deutschmann, editor, *Modeling and Simulation of Heterogeneous Catalytic Reactions*, pages 71–111. Wiley, 1 edition, November 2011.

[RS06] Karsten Reuter and Matthias Scheffler. First-principles kinetic Monte Carlo simulations for heterogeneous catalysis: Application to the CO oxidation at Ru O 2 (110). *Physical Review B*, 73(4):045433, January 2006.

[RSB94] Steven W. Rick, Steven J. Stuart, and B. J. Berne. Dynamical fluctuating charge force fields: Application to liquid water. *The Journal of Chemical Physics*, 101(7):6141–6156, October 1994.

[Sad84] Abdullah Sadiq. A new algorithm for the Monte Carlo simulation of spin-exchange kinetics of Ising systems. *Journal of Computational Physics*, 55(3):387–396, September 1984.

[SB12] Susan B. Sinnott and Donald W. Brenner. Three decades of many-body potentials in materials research. *MRS Bulletin*, 37(5):469–473, May 2012.

[SDK+10] Tzu-Ray Shan, Bryce D. Devine, Travis W. Kemper, Susan B. Sinnott, and Simon R. Phillpot. Charge-optimized many-body potential for the hafnium/hafnium oxide system. *Physical Review B*, 81(12):125328, March 2010.

[Ser11] Santiago A. Serebrinsky. Physical time scale in kinetic Monte Carlo simulations of continuous-time Markov chains. *Physical Review E*, 83(3):037701, March 2011.

[TAB+22] Aidan P. Thompson, H. Metin Aktulga, Richard Berger, Dan S. Bolintineanu, W. Michael Brown, Paul S. Crozier, Pieter J. In 'T Veld, Axel Kohlmeyer, Stan G. Moore, Trung Dac Nguyen, Ray Shan, Mark J. Stevens, Julien Tranchida, Christian Trott, and Steven J. Plimpton. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, 271:108171, February 2022.

[TG80] Donald G. Truhlar and Bruce C. Garrett. Variational transition-state theory. *Accounts of Chemical Research*, 13(12):440–448, December 1980.

[TGK96] Donald G. Truhlar, Bruce C. Garrett, and Stephen J. Klippenstein. Current Status of Transition-State Theory. *The Journal of Physical Chemistry*, 100(31):12771–12800, January 1996.

[UMGV05] Blas P. Uberuaga, Francesco Montalenti, Timothy C. Germann, and Arthur F. Voter. Accelerated Molecular Dynamics Methods. In Sidney Yip, editor, *Handbook of Materials Modeling: Methods*, pages 629–648. Springer Netherlands, Dordrecht, 2005.

[Ver67] Loup Verlet. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Physical Review*, 159(1):98–103, July 1967.

[VK11] N. G. Van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier Science, Burlington, 3rd ed edition, 2011. OCLC: 880740151.

[VMG02] Arthur F. Voter, Francesco Montalenti, and Timothy C. Germann. Extending the Time Scale in Atomistic Simulation of Materials. *Annual Review of Materials Research*, 32(1):321–346, August 2002.

[Vot07] Arthur F. Voter. Introduction to the Kinetic Monte Carlo Method. In Kurt E. Sickafus, Eugene A. Kotomin, and Blas P. Uberuaga, editors, *Radiation Effects in Solids*, NATO Science Series, pages 1–23, Dordrecht, 2007. Springer Netherlands.

[VSA+14] C. Verdon, O. Szwedek, A. Allemand, S. Jacques, Y. Le Petitcorps, and P. David. High temperature oxidation of two- and three-dimensional hafnium carbide and silicon carbide coatings. *Journal of the European Ceramic Society*, 34(4):879–887, April 2014.

[WCC+20] Junxia Wen, Rui Cao, Hongyan Che, Hao Dong, Haiyan Zhang, Yingjie Yan, Yanfei Gao, and Peter K Liaw. The oxidation effect on the cracking behavior of a Co-based alloy under thermal shocks. *Corrosion Science*, 173:108828, August 2020.

[WS33] E. Wigner and F. Seitz. On the Constitution of Metallic Sodium. *Physical Review*, 43(10):804–810, May 1933.

[XH08] Lijun Xu and Graeme Henkelman. Adaptive kinetic Monte Carlo for first-principles accelerated dynamics. *The Journal of Chemical Physics*, 129(11):114104, September 2008.