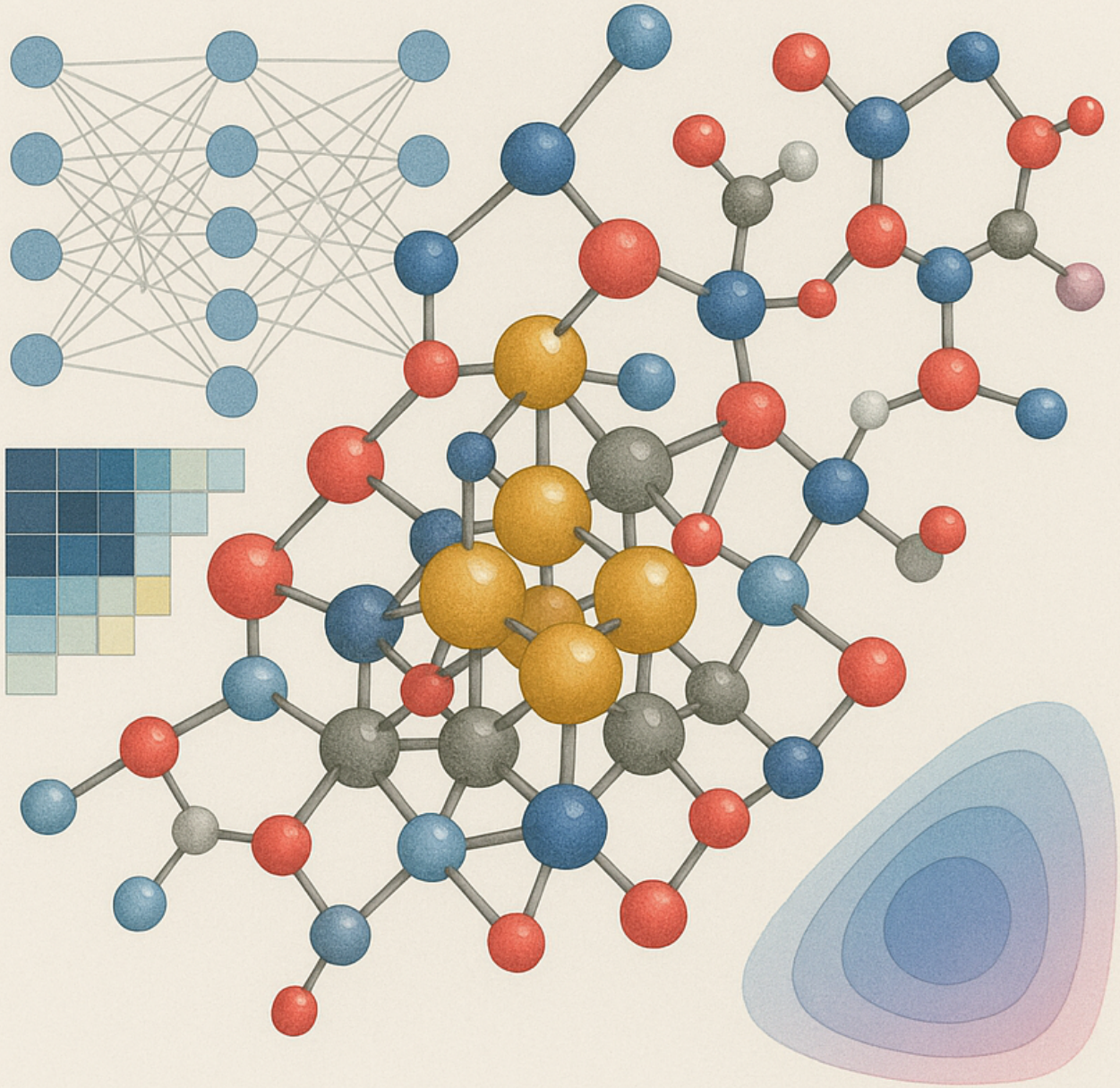
Master Thesis

Machine Learned Interatomic Potentials for Prediction of Nanocatalysts Structures

Master in Materials Science and Engineering

Hector Gerardo Diaz Nieto

Delft University of Technology



Master Thesis

Machine Learned Interatomic Potentials for Prediction of Nanocatalysts Structures

by

Hector Gerardo Diaz Nieto

to obtain the degree of Master of Materials Science and Engineering

at the Delft University of Technology,

to be defended publicly on Monday August 25th

Student number:	5918189	
Project duration:	April 11, 2025 – August 25, 2025	
Thesis committee:	Dr. K. Rossi,	TU Delft, supervisor
	Prof. dr. E. Pidko,	TU Delft (TNW)
	Dr. S. Kumar,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`. Github Repository is available at **https://github.com/Staphalon20XX/MACE-Thesis**.

TU Delft

Acknowledgments

I would like to begin by extending my heartfelt thanks to the Materials Science and Engineering department at TU Delft. Looking back, I realize I could not have made a better choice for continuing my academic journey. This experience has been truly eye-opening, enriching me not only in knowledge but also in social skills and professional connections. I am grateful to all the MSE faculty professors who have helped me grow, and who have shown me that I am capable of taking on challenges I once thought were beyond my reach.

I am especially grateful to Dr. Kevin Rossi for his continuous and unwavering commitment to guiding me throughout this past year. Under his mentorship, I have learned immensely about new developments in catalysis and machine learning — fields that continue to inspire me and that I intend to keep close in the years ahead. His support has been invaluable, and this work would not have been possible without his guidance.

I would also like to thank the ISE group at the Applied Sciences faculty for their support and constructive feedback on my thesis. Our discussions have made me appreciate the great potential of this topic and project, and I am eager to see the next steps that will emerge in this area. Special thanks go to Margareth, Sasha, Anna, and the Evgenys for their help and encouragement.

Finally, and certainly not least, I want to thank my family and friends who have supported me throughout this incredible chapter of my life. Ten years ago, I could not have imagined living, studying, and working in Europe — let alone in the Netherlands, which I have now come to know as "thuis." My deepest gratitude goes to my friends back home in Honduras, to my friends and professors in Mexico who guided me during my Bachelor's in Nanotechnology, and to the amazing people I have met here in Europe. My commitment, discipline, and dedication are the direct result of the love and support I have received from everyone I am proud to call family.

Y sobre todo, gracias, Mamá, Papá y Mónica, por haberme ayudado a llegar hasta aquí. Nada de esto sería posible sin ustedes. Gracias.

Abstract

Catalysts are everywhere. They help accelerate and enable essential industrial processes towards success by being selective towards certain products and producing higher yields of said products at an agile manner. Nowadays, the relevance of catalysts is not only in industrial production, but also in the development of novel structures which are able to provide reaction pathways of relevant processes, such as the production of green hydrogen, or conversion of CO_2 into relevant products. For this, computational simulations are used as the first step in screening potential candidates that are able to provide higher yields or selectivity in heterogeneous catalysis reactions. However, these simulations are mainly done through the use of DFT, which requires a high computational cost and convergence time. Machine Learned Interatomic Potentials (MLIPs) have risen as complements for DFT simulations via training and learning from DFT energies and forces data to provide a platform for molecular dynamics simulations used to study the movement and behavior of atoms and molecules over time. In this research project, an active learning loop is engineered with the purpose of automating the workflow of training, using, and fine-tuning a MLIP (in this case, MACE) for its further use in catalysis energetics calculations.

Contents

1

Introduction

Fossil fuels have been the staple for societal progress and industrial evolution over the past two centuries. This is mainly due to their affordability, abundance, and availability. By now, it is well known that the use of fossil fuels generates carbon dioxide and other greenhouse gas emissions, making them responsible for 75% of the worldwide greenhouse gases present in our atmosphere [1].This increasing trend (as shown in Figure 1.1) of greenhouse gas concentration in the atmosphere induces more severe climate change effects, such as intense weather patterns, higher global temperatures, and melting of the ice caps.

With global energy demand projected to increase by up to 48% by 2040, there is a growing need for sustainable alternatives in both electricity and industrial production [3]. Catalysis offers promising solutions, enabling new molecular transformations and synthetic routes, such as green hydrogen production, carbon capture, and ammonia synthesis [4]. Catalysts, which are materials that accelerate a reaction without being consumed, are crucial for these processes, accounting for over 90% of all compounds in the chemical industry [4]. These materials not only increase reaction rates but also improve selectivity and yield, reducing unwanted byproducts [5]. The ongoing search for new catalysts focuses on developing sustainable, economically feasible, and efficient materials with long lifetimes and resistance to poisoning to meet the demands of various industries [5, 6].

Nanocatalysts show great promise towards creating a path for sustainable catalysis. Due to their increased surface area compared to the bulk, nanocatalysts are able to improve reaction speeds and selectivity. Their performance may also be optimized via the adjustment of their physical features, such as shape, size, structure, content, and materials. Plenty of research has shown resource efficiency, energy consumption, and waste reduction with the use of nanocatalysts [5, 7, 8]. Metallic nanoparticles (mNPs) must be highlighted since there are studies showing potential to tune their catalytic performance by modifying these physical features. This is easier said than done, since many calculations and simulations must be done to comprehend the relation between catalytic activity, kinetics, and reaction mechanism with the structure, shape, and size of the nanostructures [9–11].

Currently, machine learning and AI models have been deployed to assist in the optimization of catalysts to understand the structure-activity relationship of the catalysts, where high-throughput catalyst screening has been the most useful in showing predictions of computable physical properties (known as descriptors), such as adsorption or formation energies, which can be related to the activity or stability of the material [12, 13]. In spite of multiple accomplishments, there are still significant challenges that must be addressed for this field to reach its full potential.

This study introduces an approach for generating machine-learned interatomic potentials (MLIPs) using

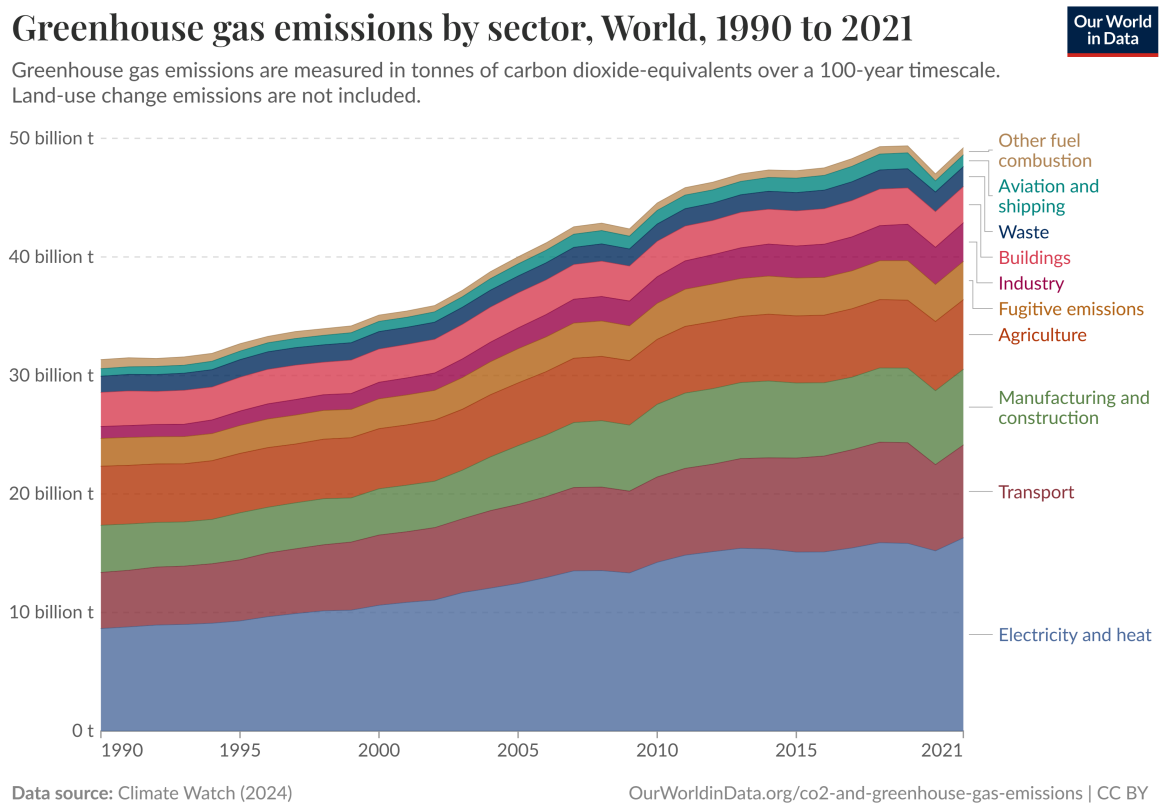


Figure 1.1: Trends in Greenhouse Gas Emissions by Sector, obtained from [2]

active loop engineering based on the MACE machine learning model to better fit the generated models for prediction at a DFT level. The objective is to create highly accurate and computationally efficient potentials for use in molecular dynamics simulations.

We've applied these potentials to investigate the stability of a Ni/Ga_2O_3 catalyst, a promising and selective material for the hydrogenation of CO_2 to form methanol [14]. CO_2 hydrogenation provides a utilization pathway of captured CO_2 from the atmosphere or from industrial processes to produce methanol, which is a highly utilized substance in the chemical industry [15]. Furthermore, the MLIPs are used to calculate the energetic barrier for the formation of the formate ion, a critical intermediate in the CO_2 hydrogenation reaction.

By demonstrating that MLIPs enable molecular dynamics simulations with both high accuracy and speed, this work suggests that they can serve as a substitute or complement to traditional Density Functional Theory (DFT) calculations in catalyst screening. This approach offers the potential to significantly accelerate the discovery of new catalysts by requiring only a fraction of the time to converge compared to conventional methods.

2

Theory and Background

2.1. Heterogeneous Catalysis

As explained by Ostwald [16], catalysis involves the acceleration of a sluggish chemical process through the presence of an external substance. The primary function of a catalyst is to reduce a reaction's activation energy, which is achieved when reactant molecules attach to the catalyst. Although the catalyst participates in the reaction, it is not consumed during the process [16]. Figure 2.1 provides a straightforward illustration of how a catalyst impacts a given reaction. The activation energy is anticipated to be lower, meaning the reaction requires less energy to proceed. Additionally, Figure 2.2 presents a schematic showing the interaction of a catalyst within a reaction, where molecules 'A' and 'B' are reactants and 'P' is the product. Initially, reactants adsorb onto the catalyst's surface, followed by their reaction, during which a peak in potential energy is observed, and then the product forms. Subsequently, the product desorbs from the catalyst.

Catalysis is categorized into two main types: heterogeneous and homogeneous catalysis. In heterogeneous catalysis, the catalyst is typically a solid, while the reactants and/or products exist in a different phase. Conversely, in homogeneous catalysis, both the catalyst and the reactants are in the same phase [16, 17]. Another significant form of catalysis is enzymatic catalysis, where molecules like proteins enhance the reaction rate [5].

Common heterogeneous catalysts encompass metals, metal oxides, metal salts, and organic materials. Noble metals receive particular attention because they are considered superior catalysts due to their exceptional performance in various large-scale industrial processes. These applications include ammonia oxidation, NOx reduction, catalytic cracking of crude oil, CO and hydrocarbon oxidation, and steam reforming of methane to CO + H_2 [17, 18].

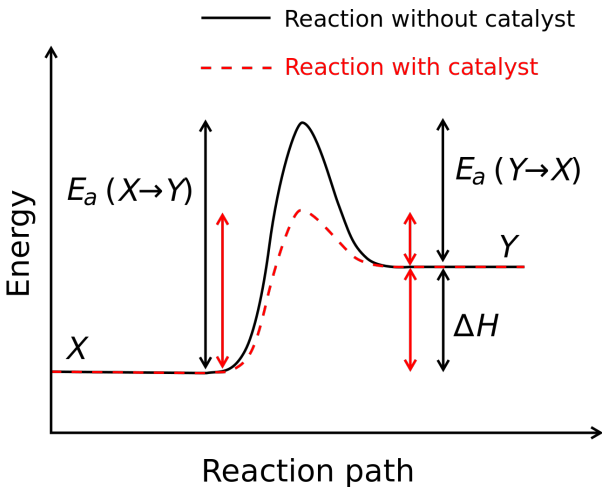


Figure 2.1: Energy reaction path with and without catalyst [18].

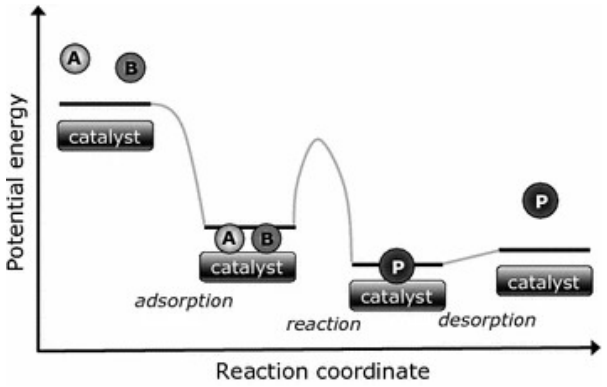


Figure 2.2: Reaction scheme with catalyst [19].

Several important characteristics of catalysts include activity, selectivity, and stability, which are defined as follows:

- *Activity: The capacity of a catalyst to accelerate a reaction's rate by lowering the energy barrier between reactants and products [16].*
- *Selectivity: The ability of a catalyst to favor the formation of a single, desired product over alternative products [20].*
- *Stability: The sustained effectiveness of a catalyst over time, coupled with its resistance to deactivation or poisoning [21].*

Currently, the Sabatier Principle serves as a guiding concept for screening new catalyst materials or for explaining the properties of existing catalysts. The Sabatier principle of optimum adsorption posits that *the binding energy between the catalyst material and the reactant(s) should be neither excessively weak nor excessively strong*. If the bond is too weak, the catalyst material and reactant might not interact effectively; conversely, if the bond is too strong, the reactant will not desorb from the catalyst surface, thereby hindering subsequent reactions [22].

The Sabatier principle is commonly illustrated through volcano plots. A qualitative depiction of the Sabatier principle is shown in Figure 2.3, where an ideal catalyst material is situated at the peak of the volcano-shaped curve. These plots are generated by comparing a measure of the reaction rate on a

catalyst against its adsorption energy. This yields a volcano-shaped curve, indicating that the reaction rate is low at both high and low adsorption strengths [22].

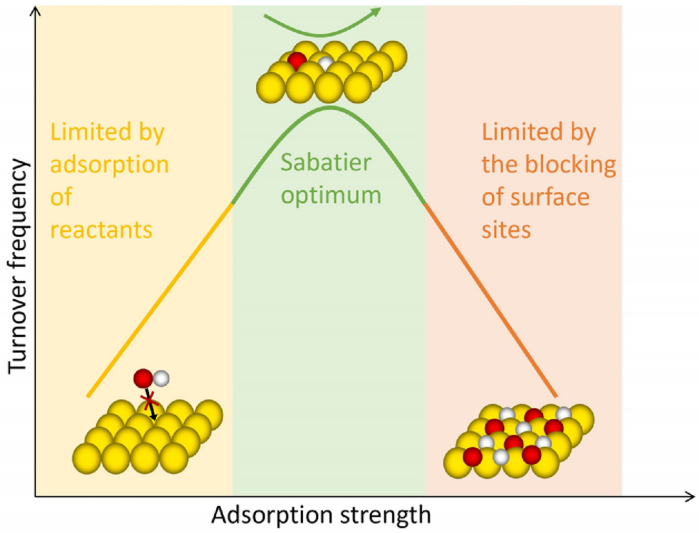


Figure 2.3: Sabatier principle represented in a volcano plot [22].

2.1.1. Nanocatalysis

Nanomaterials are simply defined as materials with a size ranging from 1-100 nm. These materials have gained recognition across numerous scientific disciplines, including healthcare, food processing, construction, cosmetics, energy manufacturing, and, notably, catalysis, owing to their distinctive properties. A nanocatalyst can be described as a nanomaterial employed as a catalyst in a chemical reaction. These materials offer unique advantages compared to conventional catalysts, such as improved mixing with reactants, a higher surface-to-volume ratio, easier separation from the reaction mixture, and greater ease in tuning catalytic activity and selectivity. The latter is emphasized as one of the most critical characteristics of nanocatalysts, as the tuning of activity and selectivity is achieved by customizing the chemical and physical attributes of the nanomaterial [8, 23]. Other benefits, as highlighted by Chen et al. [24], include the following:

- ***Large surface area***
- ***High tunability in morphology***
- ***High density of edge sites***
- ***Possibility of bimetallic catalysts***
- ***Great support materials***
- ***Provide favorable reaction environment***

Nanocatalysts have demonstrated enhanced energy efficiency and reduced costs compared to traditional catalysts. Other anticipated advantages of nanocatalysts are depicted in Figure 2.4. Beyond increased catalytic activity, scaling down a material's size to the nanoscale also confers higher conductivity, reactivity, and optical sensitivity. Common nanocatalyst materials include carbon-based nanomaterials (e.g., graphene, carbon nanotubes, and fullerenes), metal nanostructures (iron, silver, gold, cobalt), metal oxide nanoparticles (zinc oxide, titanium dioxide, silicon dioxide), and other types of nanomaterials (such as quantum dots and metal-organic frameworks) [8, 24].

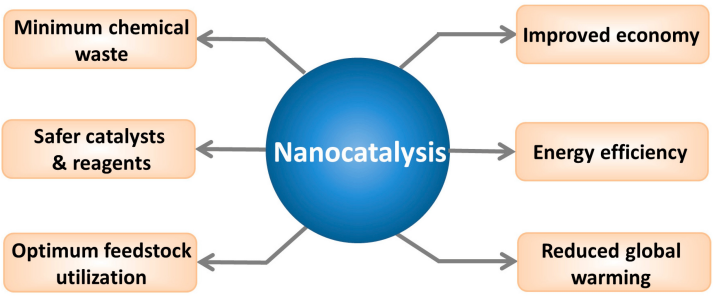


Figure 2.4: Expected benefits of Nanocatalysts [8].

Nanocatalysts employed in heterogeneous reactions have exhibited high activity, selectivity, and stability. With nanoparticles, researchers aim to enable underdeveloped reactions or optimize the activities of existing reactions. The catalytic performance of nanoparticles can be optimized by adjusting their structure, shape, size, and material composition [5]. Highly active, selective, and durable catalysts, which are in high demand for chemical manufacturing, energy conversion, and storage, can be developed with the assistance of researchers.

Since electrocatalysis is an inherent surface phenomenon, it is significantly impacted by the surface chemistry of nanostructures, which includes the density of active sites present on these structures' surfaces. Consequently, it can be concluded that the primary focus for optimizing the performance of nano-electrocatalysts used in the aforementioned electrochemical reactions can be further enhanced by tailoring their active sites. Discovering these catalysts through conventional trial-and-error methods is exceedingly time-consuming and resource-intensive; therefore, alternative optimization techniques for novel nanocatalysts must be considered [25, 26].

In nanocatalysis, identifying catalytic sites is crucial for a deeper understanding of electrocatalytic reactions and, naturally, for establishing improved criteria for future nanocatalyst design. For metal-based nanoparticles, their compositions, crystal structures, and exposed crystal facets generally dictate the properties of their catalytically active sites. Alloying is, thus, a common method to explore different activities of catalytic sites and also an effective strategy to reduce the quantity of noble metals required in the catalyst, thereby lowering material costs [27].

In metal nanocatalysts, various types of active sites can play significant roles in the material's activity. Figure 2.5 presents some examples. Firstly, unsaturated coordination sites, such as terraces, steps, kinks, and the edge and corner sites of different materials, are well-known active sites in metals involved in various electrocatalytic reactions. The coordination number of surface atoms is directly linked to their adsorption energies; consequently, atoms with lower coordination numbers (including those located at the previously mentioned active sites) are more thermodynamically unstable and potentially more active for adsorbing reaction intermediates compared to coordinated, saturated atoms [27]. Surface defects, such as vacancies and grain boundaries in metals and metallic compounds, also represent potentially highly active sites.

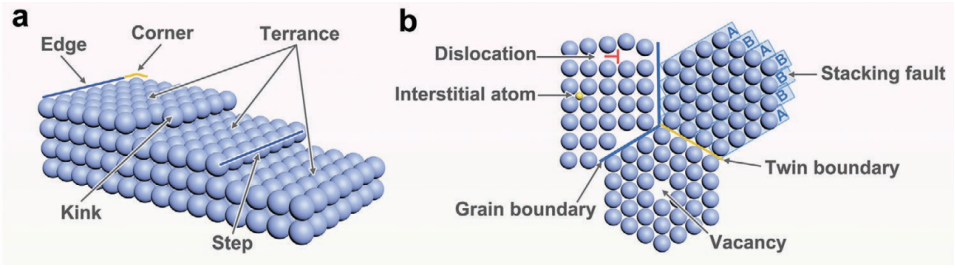


Figure 2.5: Typical active sites in metal nanocatalysts. a) Unsaturated coordination sites b) defect sites [27]

2.2. Research Methodologies in Catalysis

Studying catalysts and especially nanocatalysts is no easy chore. Decision-making takes place the moment one starts to think about which catalyst could have the potential to increase a certain reaction's activity, selectivity, or stability. From complex and multiple reaction pathways, diverse synthesis methods, multiple characterization techniques to choose from, differences in operando conditions versus simulated, stable conditions, there are a great number of factors that must be taken into account to fully understand how a specific material system works under a catalytic reaction. This can get to be very overwhelming, from economic, time, and resources perspectives [6, 28, 29].

A traditional catalysis design methodology (as seen in figure 2.6) can be divided into three major steps: Idea development and Synthesis of catalyst; Characterization, reaction mechanism, and kinetics, and stability testing; and finally, scale-up to plant reactors for industrial production. In the primary screening, through deep research in literature, state-of-the-art catalyst research, and also previous experience, researchers can come up with ideas for novel catalysts tailored ad hoc for specific applications that researchers are interested in studying (e.g., electrocatalysis, photocatalysis, bio-catalysis, etc.) [4–6]. According to the desired properties that the catalyst requires for the application at hand, the composition, shape, synthesis process, and reaction conditions are defined. At this point, researchers jump into the following stage, characterization and catalyst reaction kinetics/mechanisms. To study the surface phenomena that occur in heterogeneous catalysis very well, there are two essential steps used in modern catalysis research: Spectroscopy techniques for characterization of the material, and experiments or simulations of catalysts to study their reaction thermodynamics and kinetics [28, 29].

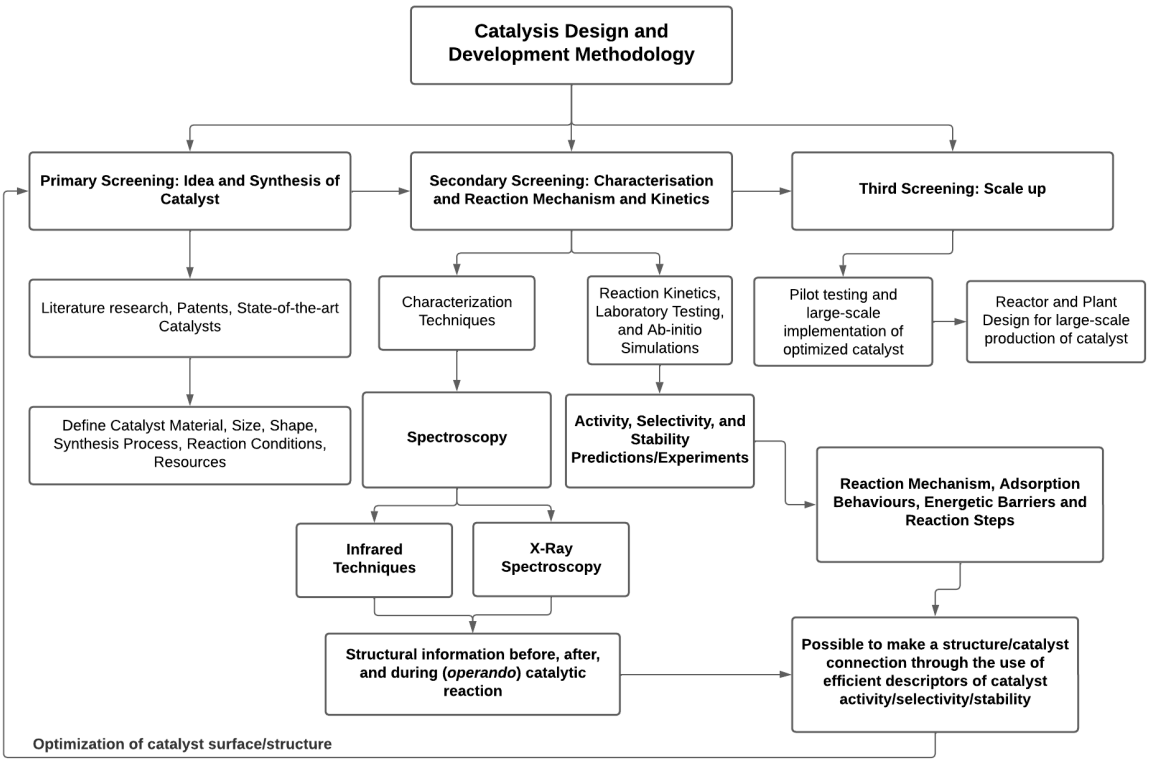


Figure 2.6: Brief and General Catalyst Screening Methodology, adapted from [28, 29]

In this research project, emphasis is placed on the second screening section of the methodology; more specifically, on the catalyst efficiency prediction. Computational methods are nowadays a staple in catalysis research, since they are able to provide information about the catalyst structure-activity relationships after doing molecular dynamics (MD) simulations. More about this will be explained in the following sub-sections [30].

2.2.1. DFT for Catalytic Research

Density Functional Theorem (DFT) is a computational method widely used due to the compromise between accuracy and computational costs. It is used for *ab initio* calculations of the structure of atoms, molecules, crystals, surfaces, and their respective energetic interactions [31]. DFT has refined the knowledge theory of fundamental chemical reactions, surface science, and electrochemical reactions [32]. In catalysis, DFT has been regularly used to accurately calculate adsorption energies, which are then used as descriptors of the catalytic activity for a large number of systems [30].

Models of reactivity trends that are able to single out catalytic activity or selectivity via important parameters are essential prerequisites for understanding and tailoring surfaces with specific catalytic properties. In principle, the kinetics of a given catalytic reaction can be understood if one calculates all reaction free energies and activation free energy barriers as a function of coverage and surface structure [33].

However, this would be extremely time-consuming and computationally demanding to calculate the reaction free energies for all energy parameters for all elementary steps under every single condition, which is why micro kinetic models are developed to evaluate catalytic properties. These models are applied to understand surface reactivity dynamics of catalysts, pinpointing the most important micro-

scopic material properties of catalysts. Reaction mechanisms, energy correlations of adsorbate energy of intermediates and transition states may be calculated, leading to the development of linear scaling relationships [34].

The presence of these relationships greatly simplifies the number of independent catalytic parameters needed to fully describe the thermodynamics of complex surface reaction networks. The emergence of empirical volcano plots over the past 50 years predicted this variable reduction by linking reaction rates in heterogeneous catalysis to one or two parameters. Meanwhile, scaling relationships have provided a theoretical basis for deriving volcano plots from first-principles DFT calculations [34].

Another linear correlation worthy of emphasis is the Brønsted-Evans-Polanyi (BEP) relation, frequently employed to characterize surface reactions on catalysts [35]. The BEP relation establishes a linear correlation between the activation energy and the enthalpy change of an elementary reaction. Stated differently, it represents a connection between the adsorption energy and the transition state energy for a given reaction [33]. This relationship enables a quantitative understanding of the volcano curve observed in heterogeneous catalysis systems [36]. The BEP relation provides an empirical approach to estimate kinetic parameters from thermodynamic values. With the aid of Density Functional Theory (DFT), activation energies and enthalpy changes for specific elementary reactions can be derived from first-principle calculations [36].

2.3. DFT Theory

As seen in previous sections, DFT usage is a staple for catalysis research due to its highly accurate energy calculations through molecular and periodic simulations. DFT is the most commonly used computational method since it provides an optimal balance between accuracy and computational cost, as compared to other semi-empirical methods and wavefunction-theory-based alternatives [30]. The main advantage of DFT over these methods is that it does not require significant *a priori* information of the system. With DFT, a wide range of catalytic features and properties can be more easily researched, due to the possibility of calculating adsorption energies, activation energy barriers, and obtaining plenty of electronic structure information [30].

DFT is based on the theory of electronic ground state structures based on the electron density, $\rho(r)$. The electron density is a physical observable, and its integration over all space offers a calculation of the total number of electrons N, as shown in equation 2.1. The electron density of a system uniquely determines the ground state energy and properties of a system, since this functional is a function of space and time. The electron density at a specific position in space can be written in terms of the individual electron wave function (see equation 2.2).

$$N = \int \rho(\mathbf{r})\mathbf{dr} = N \int ... \int |\Psi(\mathbf{r}, r_2, ...r_N|2dr_2, ..., dr_N \tag{2.1}$$

$$\rho(\mathbf{r}) = 2\sum_i \psi_i^*(\mathbf{r})\psi_i(\mathbf{r}) \tag{2.2}$$

The theoretical basis of DFT lies on two fundamental mathematical theorems proved by Kohn and Hohenberg. The first theorem states that ”the ground-state energy is a unique functional of the electron density”. This could also be interpreted as the ground state electron density is used as a basis to calculate all other properties, including energy and wave function, of the ground state. The second property describes an important property of the functional. Here, it is stated that ”the electron density minimizes the energy of the overall functional is the true electron density, corresponding to the exact solution of the Schrödinger equation” [30]. If the ”true” functional form was known, it is possible to vary the

electron density until the energy from the functional is minimized. This minimization in energy would then result in the finding of the relevant electron density.

This discussion emphasizes that the density determines the external potential, which then determines the Hamiltonian and, in consequence, the wave function. With these last two factors, the energy can be computed. Yet, there is still the issue of solving the Schrodinger equation, which arises due to the impossibility of describing the electron-electron interaction in the Hamiltonian. To solve this bottleneck, Kohn and Sham figured out that if the Hamiltonian operator for a non-interacting system of electrons is employed [30]. This way, a ”fake” starting point system of non-interacting electrons is employed. In this starting point, non-interacting electrons possess a ground-state density which is similar to some real systems of interest of electrons that do interact. Afterwards, the energy functional is divided into specific components to facilitate analysis. The complete functional is shown in equation 2.3. Here, the terms $\Delta T[\rho(r)] + \Delta V[\rho(r)]$ are joined together into the term E_{xc}, referred to as the exchange-correlation energy (see equation 2.4. It includes the quantum mechanical exchange correlation effects, and the correction for the delta in kinetic energy between the ”fake” non-interacting system and the real system [30].

$$E[\rho(r)] = T_{ni}[\rho(r)] + V_{ne}[\rho(r)] + V_{ee}[\rho(r)] + \Delta T[\rho(r)] + \Delta V[\rho(r)] \tag{2.3}$$

which may be simplified into:

$$E[\rho(r)] = T_{ni}[\rho(r)] + V_{ne}[\rho(r)] + V_{ee}[\rho(r)] + E_{xc} \tag{2.4}$$

After a comprehensive understanding of how the energy functional is represented, it is important to choose a correct computational protocol to run DFT successfully and provide accurate results. This accuracy is also dependent on the selected catalytic model. Firstly, appropriate basis sets must be inputted to describe the shape of the orbitals in an atom. There are many available basis sets to choose from according to the specific requirements of the desired calculations or working system [30]. These basis sets mathematically represent the distribution of electrons in the system of interest [37].

2.3.1. Ab-initio Molecular Dynamics

Classical MD typically employs predefined empirical potentials. In conventional MD calculations, the force acting on an atom is the result of the gradient of the energy, which depends on all nuclear coordinates. Their oversimplified representations of interatomic interactions, however, result in the generation of incomprehensible empirical force fields that are not able to correctly describe the formation and breaking of chemical bonds. Atoms in a chemical reaction or during diffusion move in a part of the configuration space which is not included in the original adjustment of the parameters set in such MD simulations [38, 39]. Classical MD is therefore not suitable for modeling catalytic reactions.

Ab initio MD (AIMD) has risen as a solution for this bottleneck. AIMD assesses interatomic forces through precise electronic structure calculations. The forces acting on the atoms are derived in each time step from a quantum mechanical theory of electron distribution given within DFT (such as GGA or LDA) [38]. Ergo, it allows the research of chemical processes, and it is able to take into consideration the contributions of various isomers of catalysts to reactions. This provides a more accurate reflection of the catalyst dynamics under realistic conditions [39].

One major benefit of applying AIMD is that it doesn’t rely on experimental data, and it is possible to accurately model the formation and breaking of chemical bonds. It allows access to electronic properties that can be related to atomic motions. However, an issue with this technique is that simulations can span

thousands or even more time steps, and performing a full self-consistent quantum calculation at every step would be computationally impractical [38].

Carr and Parrinello introduced a classical-like equation of motion for the quantum wave functions of the electrons. As seen in equation 2.5, a Lagrangian that combines the classical dynamics of the nuclei with a quantum mechanical approach of the electrons using DFT is established. This way, both nuclei and electrons are treated simultaneously in a unified framework. The combined equations of motion (shown here as $M_I\dot{\mathbf{R}}_I$) can be solved via a Verlet algorithm. These are obtained by finding the stationary point of the Lagrangian under variations of ψ_i and R_I with the orthogonality constraint (set here by the Kronecker delta δ_{ij}. To keep the electrons close to the ground state, fictitious dynamics are applied, which propagate initially a fully minimized set of orbitals to subsequent minima corresponding to each new nuclear configuration [38, 40]. Using a fictitious electron mass represented as μ ensures minimal energy transfer from nuclei to electrons.

$$\mathcal{L} = \frac{1}{2}\left(\sum_{I}^{\text{nuclei}} M_I\dot{\mathbf{R}}_I^2 + \mu \sum_{i}^{\text{orbitals}} \int d\mathbf{r}\left|\dot{\psi}_i(\mathbf{r},t)\right|^2\right)$$
$$-E[\{\psi_i\},\{\mathbf{R}_I\}] + \sum_{ij}\Lambda_{ij}\left(\int d\mathbf{r}\psi_i^*(\mathbf{r},t)\psi_j(\mathbf{r},t) - \delta_{ij}\right) \tag{2.5}$$

In AIMD, all electrons are represented. Including all core electrons with their respective basis sets for heavy atoms is computationally expensive. Since core electrons are usually not of interest, they may be replaced by potential functionals in the Hamiltonian, which are known as effective core potentials (ECPs), where the electron-electron repulsion of the replaced core orbitals is included. These effective core potentials are also known as pseudo-potentials, and they are developed considering an isolated atom of one element [30].

In periodic, crystalline systems, plane-wave (PW) basis sets are more suitable. Here, the electrons in a band are described by orbitals expanded in a basis set of plane waves. Here, each Bloch state is expressed as a Fourier series, whose basis functions are computed efficiently and straightforwardly. However, the inclusion of all core electrons for heavy atoms in PW-based calculations is quite expensive. In order to reduce this computational cost, pseudopotentials (PPs) are used to smear the nuclear charge and model the electron cores. Each of these PPs contains the minimum energy cutoff, which might be used in the calculations. Different types of PPs are available to use, where the ultrasoft pseudopotentials (USPPs) are widely used due to their significantly lower cutoff energy values. This introduces lower accuracy in the system; however, empirical parameters are introduced to balance this tradeoff. This, however, limits the transferability of these PPs. Nonetheless, the implementation of the projector augmented-wave (PAW) method coupled with USPPs avoids this transferability disadvantage, providing nearly identical results which are also congruent with those from all-electron calculations [30].

In this thesis project, Vienna Ab Initio Simulation Package (VASP) is used to perform ab initio quantum mechanical calculations. Here, minimization of forces and geometry optimization are performed, as well as MD simulations. VASP calculates an approximate solution to the many-body Schrodinger equation to find the electronic ground state. This may be done using DFT by solving the Kohn-Sham equations, or with the Hartree-Fock method. Essential characteristics of the tested system, such as electron orbitals, charge density, and local potential, are all represented using a plane-wave basis set within the aforementioned PAW method. Using PAW pseudopentials, valence electrons may be efficiently modeled while accurately describing the behavior near atomic nuclei [41].

2.3.2. Exchange Correlation Energy Functional

As seen in equation 2.4, one major task in the Kohn-Sham theory is to derive approximations to the exchange-correlation energy functional. The choice of the functional form for E_{xc} is what mainly differentiates various DFT methods. The true form of E_{xc} is not known. Multiple functionals have been developed in the last decades to derive approximations of this factor. Two types of functionals exist: non-empirical and empirical. The former satisfies certain constraints to make use of the known exact constraints on the true Kohn-Sham functional. The ladder makes use of parameters that are obtained by experimental or ab initio data [30].

Different functionals have different levels of accuracy, based on the way the Schrodinger equation is solved within these functionals. Perdew and Schmidt have developed a categorization of different functionals according to their accuracy level, named Jacob's ladder (see fig 2.7). Higher levels of this ladder represent higher accuracy levels, with the highest level being the resolution of the Schrodinger equation without using approximations. The chemical heaven is reached once the accuracy needed to predict the rates of chemical reactions is acquired (where energy errors of the order of 1 kcal/mol or 0.0434eV (or lower) are desired) [42].

The first level is the local density approximation (LDA), which assumes variations of the density to be sluggish and treats the local density as a uniform electron gas. Above the LDA lies the generalized gradient approximation (GGA), where information about the local electron density and local gradient in the electron density is included. The meta-GGA functionals include the Laplacian (second derivative) of the density. Above this, the hybrid functionals (aka GGAs) are located. These combine non-local, exact exchange-energy density with a GGA [30, 42]. It is important to keep in mind that an increase in accuracy or step in this ladder also imposes a higher computational cost on the DFT calculations due to the added complexity of the functionals. This proves to be a major bottleneck when using DFT, due to the constant increase in the complexity of the novel catalyst design.

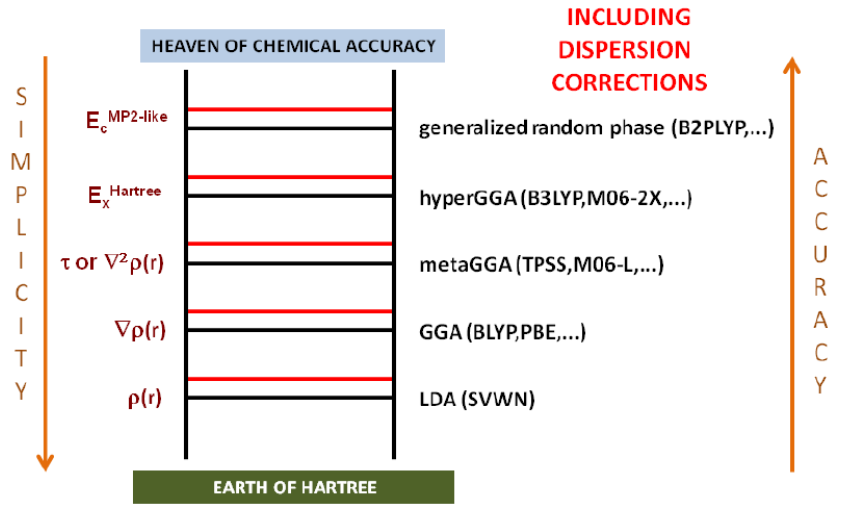


Figure 2.7: Jacob's Ladder of DFT functional approximations for the Exchange Correlation Functional [42, 43]

Choosing the right functional for a given system is crucial yet challenging, as no single functional works universally well. The accuracy of results and computational expense heavily depend on this selection. Therefore, a thorough understanding of the similarities and differences between commonly used functionals is essential. Researchers actively debate the optimal functional, even for seemingly straightforward systems like transition metal surfaces and complexes, highlighting the complexity of this decision [30]. Two drawbacks that are present in DFT approximations are the self-interaction error (SIE) and

the missing long-range correlation effects. The SIE errors are not present in the Hartree-Fock methods, since the self-interaction energy is cancelled out by the contributions to the energy from exchange. If the exact Kohn-Sham functional were known, this would also apply here. Hybrid functionals assist in reducing SIE due to a fraction of approximate DFT exchange being replaced by Fock exchange.

A novel method named DFT+U or Hubbard method is a computationally low-cost method where underestimated electronic interactions are corrected by adding the semi-empirically tuned numerical parameter "U". This method is, in short, a combination of HF and DFT. It takes into account the on-site Coulombic repulsion among localized d electrons by incorporating an energetic penalty for delocalization. A correction for the self-interaction to the DFT energy is added by introducing the numerical parameters "U" and "J", which involve different aspects of self-interaction [30]. DFT+U methods have been reported to be useful for the reproduction of the experimental trends observed in the oxygen evolution reaction (OER) and hydrogen evolution reaction (HER) activity of catalysts [30].

The Perdew-Burke-Ernzerhof (PBE) and Becke, 3-parameter, Lee-Yang-Parr (B3LYP) functionals are prominent functionals widely applied in solid-state calculations. While each provides a unique formulation for the exchange-correlation energy (see equations 2.6 and 2.7), B3LYP stands out as the most commonly used hybrid functional, renowned for its excellent reproduction of equilibrium geometries across the periodic table. Similarly, PBE0 (see eq 2.8), a hybrid derivative of PBE that includes Hartree-Fock exchange energy in its exchange-correlation calculation, generally provides improved performance for geometrical features [30].

$$E_{\text{xc}}^{\text{B3LYP}} = (1-a)E_{\text{x}}^{\text{LSDA}} + aE_{\text{x}}^{\text{HF}} + b\Delta E_{\text{x}}^{\text{B}} + (1-c)E_{\text{c}}^{\text{LSDA}} + cE_{\text{c}}^{\text{LYP}} \tag{2.6}$$

$$E_{\text{XC}}^{\text{PBE}} = \int d^3r\rho(\mathbf{r})\epsilon_{\text{XC}}^{\text{PBE}}(r_s(\mathbf{r}), s(\mathbf{r}), \zeta(\mathbf{r})) \tag{2.7}$$

$$E_{\text{xc}}^{\text{PBE0}} = \frac{1}{4}E_{\text{x}}^{\text{HF}} + \frac{3}{4}E_{\text{x}}^{\text{PBE}} + E_{\text{c}}^{\text{PBE}} \tag{2.8}$$

2.3.3. Force Calculation in DFT

Besides energy, forces can also be calculated using DFT. This is generally done via the application of the Hellmann-Feynman theorem. This theorem is defined as a principle that describes the rate of change of energy concerning a parameter in the Hamiltonian [44]. In DFT, a classical potential energy function does not exist; rather, there is a Hamiltonian $\mathcal{H}$ depending on the ionic positions $\mathbf{R}_i$ and the electronic positions $\mathbf{r}_i$. The total energy is defined in equation 2.9, where n(**r**) states the electronic ground-state density and φ_i are the Kohn-Sham orbitals. The exchange correlation energy is also included in this expression [41].

$$\begin{aligned} E_{\text{tot}} = &-\frac{1}{2}\int \sum_i \psi_i^*(\mathbf{r})\nabla^2\psi_i(\mathbf{r})\, d\mathbf{r} \\ &- \int \sum_A \frac{Z_A}{|\mathbf{r}-\mathbf{R}_A|} n(\mathbf{r})\, d\mathbf{r} + \frac{1}{2}\int\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\, d\mathbf{r}'\, d\mathbf{r} \\ &+ E_{\text{xc}} + \frac{1}{2}\sum_{A\neq B} \frac{Z_A Z_B}{|\mathbf{R}_A - \mathbf{R}_B|} \end{aligned} \tag{2.9}$$

To calculate the force acting on an ion, the Hellmann-Feynman theorem is applied. Let $\hat{H}(\lambda)$ be a Hamiltonian that depends smoothly on a parameter λ, and let $\Psi(\lambda)$ be a normalized eigenfunction of $\hat{H}(\lambda)$ corresponding to eigenvalue $E(\lambda)$, such that:

$$\hat{H}(\lambda)\Psi(\lambda) = E(\lambda)\Psi(\lambda) \tag{2.10}$$

Then, the Hellmann-Feynman theorem states:

$$\frac{dE}{d\lambda} = \left\langle \Psi(\lambda) \left| \frac{d\hat{H}}{d\lambda} \right| \Psi(\lambda) \right\rangle \tag{2.11}$$

This result holds provided that $\Psi(\lambda)$ is an exact eigenstate of the Hamiltonian and that the dependence on λ is smooth. This parameter is typically taken to be the position of a nucleus $\mathbf{R}_A$. The total energy of the system depends on the nuclear positions, and the force on nucleus A is given by:

$$\mathbf{F}_A = -\nabla_A E_{\text{tot}} = -\left\langle \Psi \left| \nabla_A \hat{H} \right| \Psi \right\rangle \tag{2.12}$$

Under the Born-Oppenheimer approximation, the electronic wavefunction Ψ adjusts instantaneously to changes in nuclear positions, and the force can be computed directly from the electrostatic interactions. For a system of nuclei and electrons, the Hellmann-Feynman force expression becomes:

$$\mathbf{F}_A = \nabla_A \sum_A \int \frac{Z_A}{|\mathbf{r} - \mathbf{R}_A|} n(\mathbf{r})\, d^3r - \nabla_A \frac{1}{2} \sum_{A \neq B} \frac{Z_A Z_B}{|\mathbf{R}_A - \mathbf{R}_B|} \tag{2.13}$$

Here:

- Z_A is the nuclear charge of atom A,
- $n(\mathbf{r})$ is the electronic ground-state density,
- $\mathbf{R}_A$ and $\mathbf{R}_B$ are the positions of nuclei A and B, respectively.

The gradient of the total energy concerning its position $\mathbf{R}_A$ is used to calculate the force. The Hellmann-Feynman theorem allows this gradient to be computed without the requirement to consider the derivation of the wave functions, which are assumed to be the exact eigenfunctions of the Hamiltonian [41, 44, 45].

2.3.4. DFT and Nudge Elastic Band in Catalysis

Catalytic reactions typically proceed via several reaction steps where the reactants are converted into products by the involvement of intermediates and transition steps (TSs), of which some are too unstable to be isolated experimentally. DFT becomes then a very prominent tool in this area, since it allows the interception of all relevant stationary points on the potential energy surfaces (PESs), which is therefore useful for the identification of the missing structures/steps and their related energy barriers. Via DFT, the most accessible pathways may be identified by determining the lowest activation barriers along with their corresponding involved rate-determining states, which is crucial for the structural engineering and electronic modification of novel catalysts [30]. Figure 2.8 demonstrates an example of how the PES can show a reaction pathway with its minima and transition state.

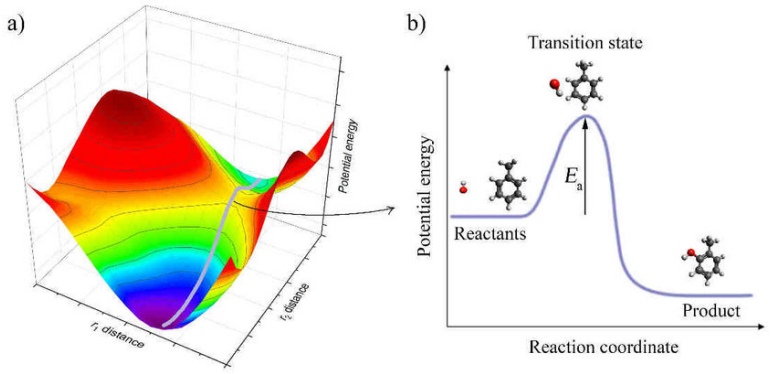


Figure 2.8: Potential Energy Surface Model. a) PES model with reaction path, minima, and transition state. b) 1D representation of PES model [46]

Once a new intermediate is found, the minimum energy path (MEP) connecting it with the starting structure may be calculated. It is important to recognize that while local geometry optimization is a common function within density functional theory (DFT) packages, the accurate determination of reaction pathways and the precise characterization of the saddle point—representing the transition state between two local minima on the potential energy surface requires the implementation of appropriate computational protocols. The existence, operational efficiency, and trustworthiness of these protocols are paramount for successful theoretical modeling in the field of catalysis [30].

The transition state theory is therefore adopted to map out the reaction network. It is based on the identification of a transition state on the PES, which corresponds to a maximum along a MEP for a particular reaction coordinate. Pre-existing algorithms and software are available for the application of this theory, for example, Gaussian [47], ORCA[48], and VASP [41]. These allow for the utilization of applied algorithms.

Nudge elastic band (or NEB) is a commonly used method to study the transition state theory, where saddle points and minimum energy paths between known reactants and products are found. The working mechanism of this technique is to optimize a number of intermediate images along the reaction path. The use defines an initial and final state, each representing the start and end of a specific reaction path. Each image then finds the lowest energy possible while trying to maintain an equal spacing to the neighboring images. In this project, NEB is coupled with the climbing image technique, where the highest energy image is driven up to the saddle point. Here, the true force at this image along the tangent is inverted, and this way the image tries to maximize its energy along the band while minimizing it in all other directions, getting the exact saddle point (and in consequence, intermediate configuration of the transition path) [49, 50]. Figure 2.9 shows an example of the NEB and NEB couple with climbing image analyses.

NEB has been previously applied to study the entire reaction pathway of CO_2 hydrogenation to methanol with a single potential in an indium oxide surface. Here, all the transition states going from the absorption of CO_2 on the system's surface up to the formation of methanol are reportedly calculated via NEB using DFT and MLIPs. These MLIPs are based on Gaussian Approximation Potentials fitted with DFT data, coupled with SOAP [51]. This study showed that after multiple active learning loops, very accurate MLIPs were able to almost completely replicate the DFT performance of transition states energy barrier calculations (see figure 2.10). Taking this into consideration, it is of great interest to observe if the MACE-based MLIPs after multiple active learning loops are able to replicate the DFT calculation between two reaction intermediates. In this case, studying the formation of the formate ion was chosen as the candidate to test this theory.

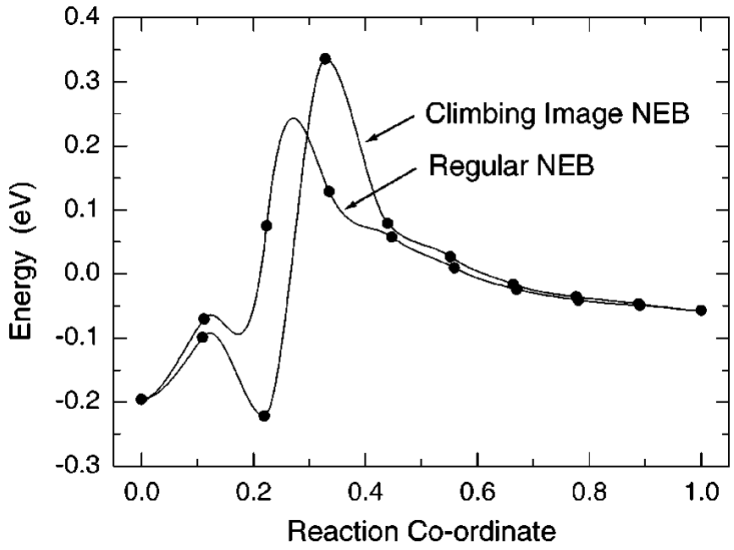


Figure 2.9: NEB and Climbing image NEB analyses examples, obtained from [49]

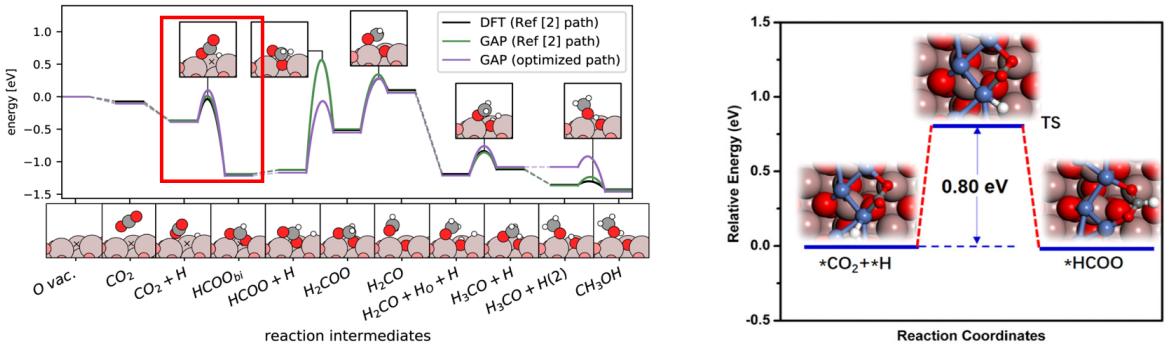
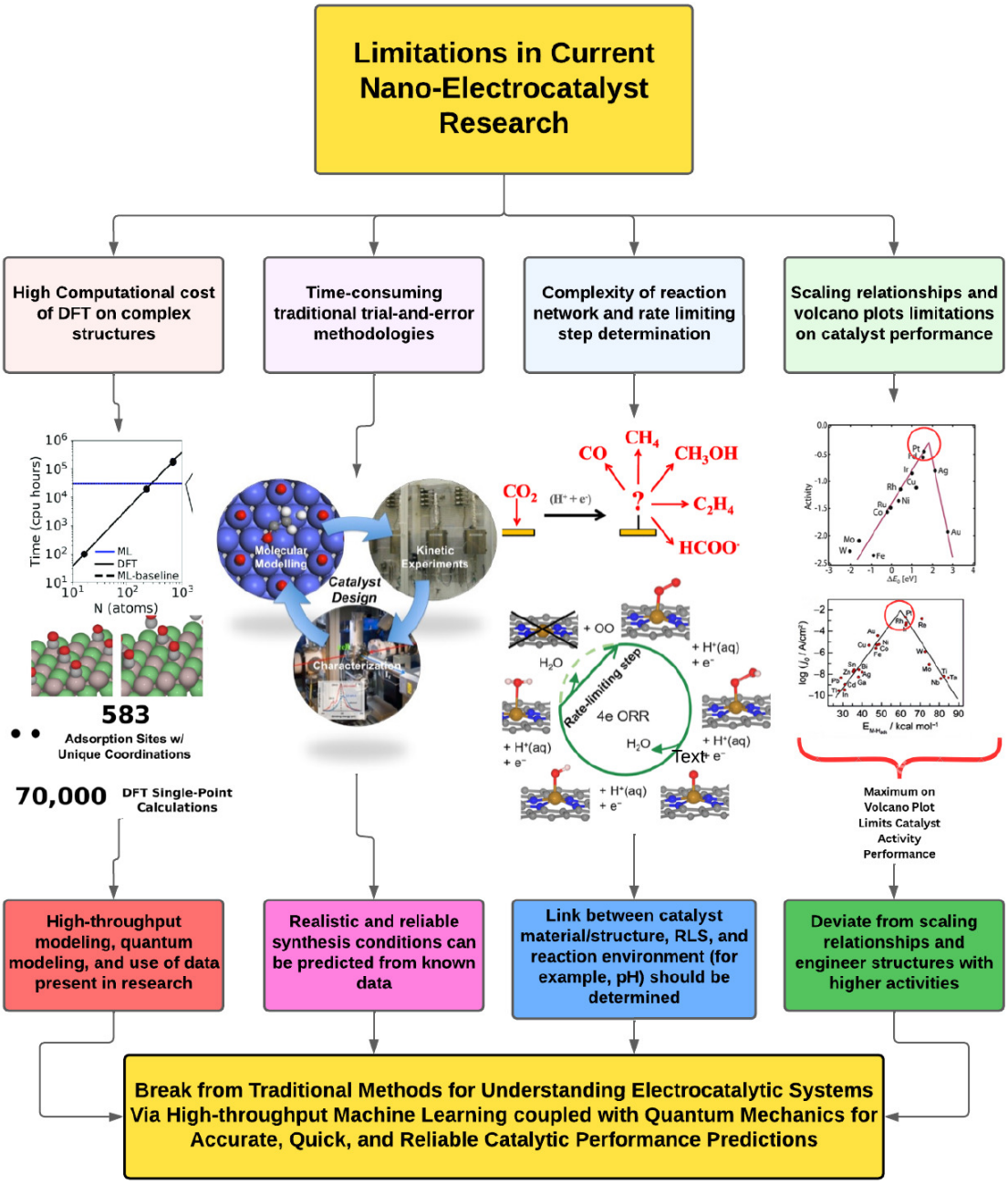


Figure 2.10: CO_2 hydrogenation (a) Full Reaction Pathway to Methanol Formation; (b) Formate ion formation transition step, obtained from [51, 52]

2.3.5. DFT Limitations

Even though great advances have been made in the field of catalysis throughout time, there are still major limitations that prevent faster and efficient screening of nanocatalysts for their application in large-scale applications. Figure 2.11 shows a general overview of the limitations found in this research area.

One of these limitations includes the dependency of DFT-based simulations for calculating different descriptors (adsorption energies, relaxation of structures, etc) of catalysts. DFT is very accurate in determining these energetic properties when done correctly; however, problems start to arise when larger, complex structures need to be simulated or materials with multiple compositions are desired to be studied. High computational power and a great amount of time would be required to obtain the entire matrix of DFT calculated results for these types of materials. DFT calculations are usually carried out in a vacuum. This ignores the complex reaction environment present in a practical electrocatalytic reaction, such as the electrolyte effect, pH, solvent effect, and others. Of course, this is extremely challenging to implement as well in DFT calculations, as explained by Liao et al. [53].



[1] A. K. Pimachev and Sanghamitra Neogi, “First-principles prediction of electronic transport in fabricated semiconductor heterostructures via physics-aware machine learning,” *npj Computational Materials*, vol. 7, no. 1, Jun. 2021, doi: https://doi.org/10.1038/s41524-021-00562-0.
[2] Z. W. Ulissi, M. T. Tang, J. Xiao, et al., “Machine-learning methods enable exhaustive searches for active Bimetallic facets and reveal active site motifs for CO2 reduction,” ACS Catalysis, vol. 7, no. 10, pp. 6600–6608, Oct. 2017, ISSN: 21555435. DOI: 10 . 1021 / ACSCATAL . 7B01648 /SUPPL{_}FILE/CS7B01648{_}SI{_}001.PDF. [Online]. Available: https://pubs.acs.org/doi/abs/10.1021/acscatal.7b01648
[3] Ghent University, “Catalyst design | Laboratory for Chemical Technology,” Ugent.be, 2025. https://www.lct.ugent.be/research/catalyst-design
[4] R. Kortlever, J. Shen, K. J. P. Schouten, F. Calle-Vallejo, and M. T. M. Koper, “Catalysts and Reaction Pathways for the Electrochemical Reduction of Carbon Dioxide,” The Journal of Physical Chemistry Letters, vol. 6, no. 20, pp. 4073–4082, Sep. 2015, doi: https://doi.org/10.1021/acs.jpclett.5b01559.
[5] S. Yu, Z. Levell, Z. Jiang, X. Zhao, and Y. Liu, “What Is the Rate-Limiting Step of Oxygen Reduction Reaction on Fe–N–C Catalysts?,” Journal of the American Chemical Society, vol. 145, no. 46, pp. 25352–25356, Nov. 2023, doi: https://doi.org/10.1021/jacs.3c09193
[6] K. Zhang, W. Guo, Z. Liang, and R. Zou, “Metal-organic framework based nanomaterials for electrocatalytic oxygen redox reaction,” Science China Chemistry, vol. 62, no. 4, pp. 417–429, Mar. 2019, doi: https://doi.org/10.1007/s11426-018-9441-4.
[7] S. M. Stratton, S. Zhang, and M. M. Montemore, “Addressing complexity in catalyst design: From volcanos and scaling to more sophisticated design strategies,” Surface Science Reports vol. 78, no. 3, p. 100 597, Aug. 2023, ISSN: 0167-5729. DOI: 10.1016/J.SURFREP.2023.100597

Figure 2.11: Current limitations found in state-of-the-art research for Nanoelectrocatalysts’ performance and design

2.4. Machine Learning for Heterogeneous Catalysis

Design of current heterogeneous catalysts has been done using trial-and-error experimental approaches that mainly rely on the chemical intuition of researchers and catalysis practitioners [12]. Identifying optimal reaction conditions, designing efficient catalysts, and revealing catalytic mechanisms all fall into this paradigm. These are usually aided by traditional experimental and computational methods, which rely mainly on prior knowledge and could be vulnerable to human cognitive biases. DFT has been a game changer over the years, of course, where simplified model systems can capture the critical aspects of complex realistic systems and therefore are able to open the window for rational design of experiments [54]. Yet, the high computational cost of using this and other types of quantum mechanical (QM) methods limits the availability of catalyst spaces that can be examined [55].

Recent progress and studies have shifted the focus towards merging Machine Learning with QM models to drive forward rational catalyst design. Machine learning (ML) refers to a branch of artificial intelligence that can lower the computational cost of complex systems and structures. Existing data is given as an input, and then a training model is generated to predict results outside of the training dataset [54]. ML models have started to gain popularity in designing and discovering catalysts. Catalyst design via predictive models is multifaceted, and catalyst performance depends on many variables, like composition, morphology, size, support material, and environment, to mention some. Due to the number of variables and the extensiveness parameter space, design and optimization of heterogeneous catalysts using traditional methods is challenging and time-consuming [12].

Machine learning algorithms generally demand significantly less processing power compared to density functional theory (DFT)-based simulations. Pimachev et al.'s research directly compared the computational costs of DFT and ML for predicting electronic transport in semiconductor hetero-structures [56]. Their findings indicate that DFT modeling time scales quadratically with system size (i.e., the number of atoms), while the ML approach shows no increase in computational demand as system size grows (see figure 2.12a). Conversely, a study by Zhang et al. developed a deep potential neural network for MD simulations. In their work, the computational processing power required for their neural network showed only a marginal difference in scaling compared to DFT-based simulations (see figure 2.12b [57].

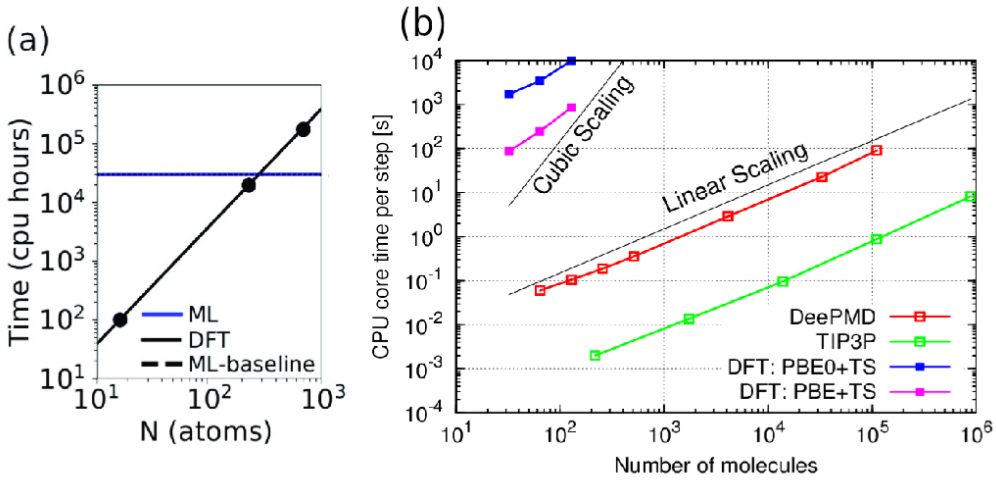


Figure 2.12: Comparison between the Computational Demand of DFT and Machine Learning for MD Simulations [56, 57]

For catalyst screening using machine learning models, a general workflow can be found in Figure 2.13.

Firstly, a dataset that contains various catalyst materials must be created, where materials with different compositions, support types, and particle sizes may be found. Each catalyst is described by its properties, such as electronic-structure properties, physical properties, and atomic properties. The important physicochemical properties of the catalyst should be captured as 'features' of the catalyst. ML tools are then used to find patterns, build models, or discover descriptors that are able to map these features describing the catalyst to their corresponding figures of merit [55].

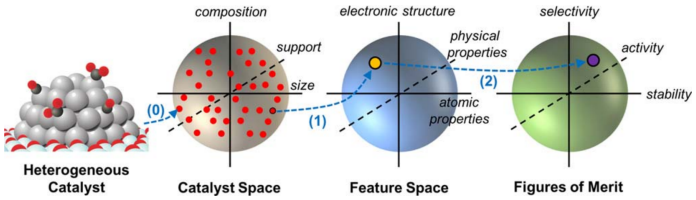


Figure 2.13: General workflow of ML catalyst screening, going from the catalyst space containing catalysts with different compositions, sizes, and support material; then, the feature space, where their properties are found. Lastly, the ML algorithms build models or find descriptors that connect their features with figures of merit established in the algorithm [55].

There exists a large number of different types of ML learning models characterized by different criteria. They can be initially classified as supervised or unsupervised learning. In supervised learning, the relation between input and output variables is provided. An example of this is regression, where an input/output relation is constructed. Meanwhile, in the latter, clustering of data using similarities in the dataset is used, where no relation between the input/output is constructed. Some common types of machine learning models are known as clustering, classification, estimation/prediction, and association [58]. There are also semi-supervised methods, which are a combination of supervised and unsupervised learning, where the model is trained using a small number of labeled samples and a high number of unlabeled samples. Finally, reinforcement learning models are trained via a reward/punishment mechanism, where desired actions are rewarded and undesired behaviors are not [59].

Some Machine Learning models that have been applied towards catalysts can be seen in table 2.1. Choosing the right model will mainly depend on the application and adoption of the problem at hand. In this research, a special focus is placed on ML learning models that have been applied towards predicting catalyst activity or descriptor behavior concerning the surface of metallic nanoparticles.

Table 2.1: Common types of ML Algorithms used for Catalyst Design

Model	Definition
Clustering Models	These models aim to group the data according to the similarities in variables. Data entries in a cluster are similar to each other due to some of their properties, while they are not similar to the constituents of the other clusters. Examples include k-means clustering, hierarchical clustering, and distribution and density-based clustering [58]
Regression/ Estimation Models	These methods model the relationship between one or more input variables with an output variable. The model then estimates the unknown output of a new set of input variables. Multiple regression, artificial neural networks, random forest regression, and support vector regression are types of estimation models. ANN is one of the most commonly used ML techniques in catalysis [58].
Classification Models	These divide the data into specific classes using values or ranges of the output variable. Examples include decision trees, k-nearest neighbor algorithm, Bayesian classification, and logistic regression [58].

2.4.1. Machined Learned Interatomic Potentials

Modern machine learning models are able to properly predict and successfully deal with physical and chemical properties of molecules, such as vibrational spectra, atomic charges, chemical potentials, and ionization potentials [60]. These machine learning models are known as Machine Learned Interatomic Potentials (MLIPs). These models have been applied in many different areas of research, such as catalysis, chemical reactivity, drug discovery, and materials design, just to mention a few. Due to the increase in size and timescales achieved in ML dynamics, more accurate modeling of complex material properties is already being made [12, 60]. Figure 2.14 shows that there is an increasing trend in the number of published articles focusing on MLIPs, where its integration into material science started approximately 15 years ago.

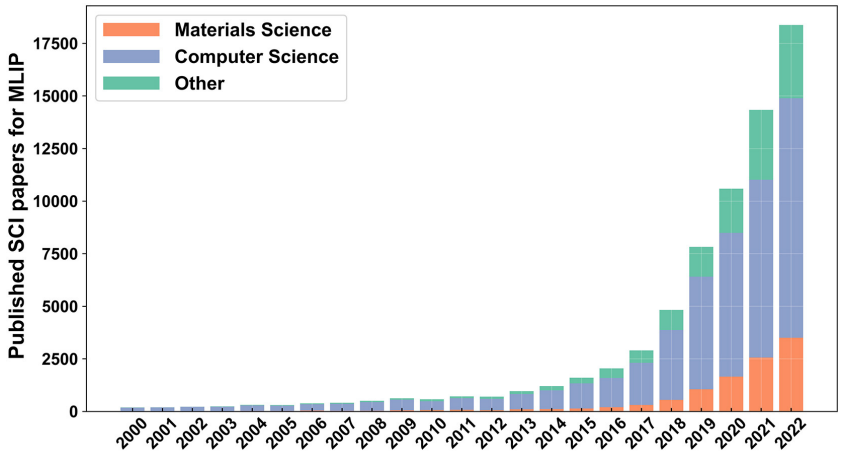


Figure 2.14: Trend in Published Research Papers Focusing on MLIPs throughout the 2000s, from the Science Citation Index [61]

One of the main reasons why ML algorithms have made such an impact in this area is due to the engineering of novel Machine Learned Interatomic Potential algorithms (or MLIPs). These models are able to replicate the complex quantum mechanical behaviors in molecular and solid materials. The engineering, design, and training/learning of these models are complex; however, various approaches, such as the one proposed by Wang et al [62], have simplified this process into several key steps. These may be seen in figure 2.15.

All steps of the development of MLIPs are essential to create a successful and accurate model; however, data collection, being the first step, is what makes all of this possible. The more diverse and clean your training data is, the more robust your model will be. The use of easily available and open datasets is, therefore, necessary to create such robust models, where biases are reduced and predictions are more accurate. Open-source material databases, such as the Materials Project or C2DB, contain a high amount of data obtained from DFT calculations [62, 63].

The most important step in creating an efficient MLIP is the descriptors or materials. Here, the conversion of spatial configurations into machine learning datasets (also known as materials' descriptors) takes place. These descriptors determine the quality of the initial dataset for the ML and define the highest accuracy the MLIP may achieve. The design of descriptors is crucial and subject to several physical constraints to ensure their robustness and accuracy [60, 62]. Additional information regarding these constraints may be found in appendix A.

After training, the process typically begins with data cleaning, pre-processing, and normalization to ensure the quality and consistency of the input data. Once prepared, an appropriate machine learning model is selected based on the specific application; common choices include Gaussian approximation models, neural networks, and active learning frameworks. The selected model is then trained using the cleaned data, during which an optimization algorithm, such as Adam, is employed to iteratively update the model parameters according to feedback from a loss function, which quantifies the model's performance on the training set. Additionally, various hyperparameters, such as batch size, learning rate, and the number of hidden layers in a neural network, may be fine-tuned to enhance performance. Following training, further improvements can be made based on evaluation results and user feedback. These may involve architectural adjustments, additional hyperparameter optimization, or the application of other model refinement techniques before final deployment [62].

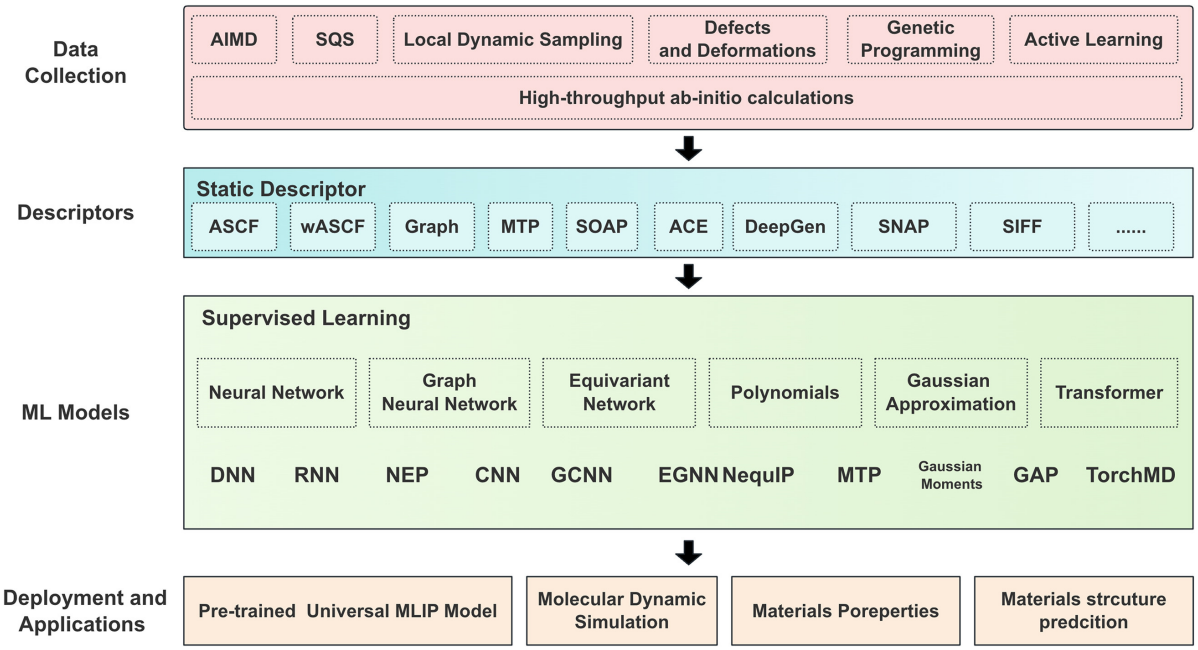


Figure 2.15: Flowchart of MLIPs Development [62].

Figure 2.16 shows a chart that contains relevant MLIPs that have been engineered during the past years. Going over all the different types of architectures and MLIPs would be out of the scope for this research project. Machine learning potential algorithms are generally classified in either kernel methods or neural networks, where in this research, neural networks will be of main focus.

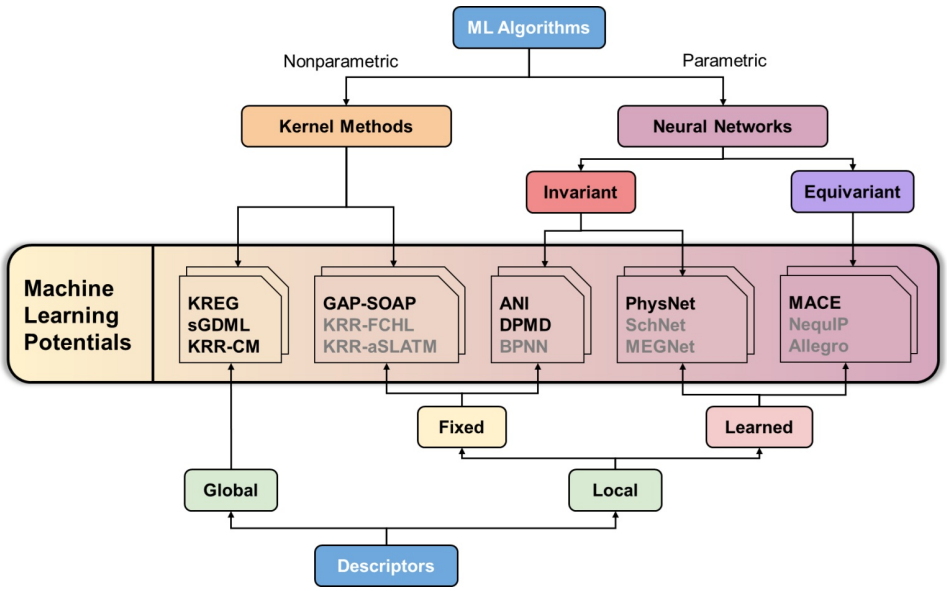


Figure 2.16: Examples of available MLIPs algorithms [64].

2.4.2. Neural Networks and Deep Learning

Neural networks are a type of machine learning algorithm designed to perform specific tasks by imitating the human brain network, with the capacity of building up its own rules of behavior, in a similar fashion as we humans do through experience. The neural network algorithm may be implemented due

to the fact that these networks are run through massively parallel computing systems made up of a large number of basic processing units (or neurons), which are interconnected and learn from their environment. The objective of the algorithm is to then modify the weights of the network in a sequential and supervised way until a specific objective (or accuracy) is reached. Here, it is also important to introduce the term deep learning. A general artificial neural network model may be seen in figure 2.17, where the input signals are represented by $x_0, x_1 ... x_p$, which are given by the user or other neurons. The weights (shown here as w) modify the received information, which can attenuate or amplify the values that are propagated into the neuron. Then, after the summing function, a bias is also placed as a threshold or intercept of the neuron. In this type of learning, the weights are modified until a certain threshold is reached [65].

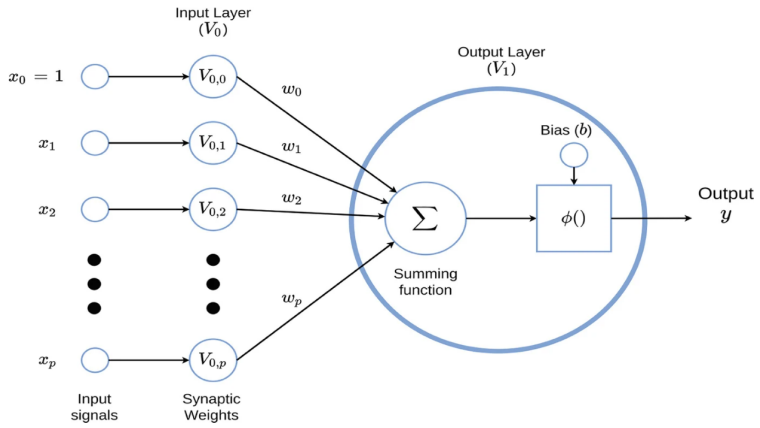


Figure 2.17: Example of Unilayer Neural Network, obtained from [65]

The example shown in Figure 2.17 depicts a unilayer neural network, which has a low processing capacity by itself, and the level of applicability is also low. Therefore, multiple neural networks should be interconnected to enhance the learning ability of the artificial neural network. An applied example of this is deep learning (DL) models. These models are artificial neural networks that utilize more than one hidden layer; in other words, more neurons are used for the implementation of the model. They are also able to capture nonlinear aspects of complex data better, but of course, more computing power is required to make these models work. Figure 2.18 shows an example of an artificial deep neural network with multiple input variables, two hidden layers, and four output variables [65].

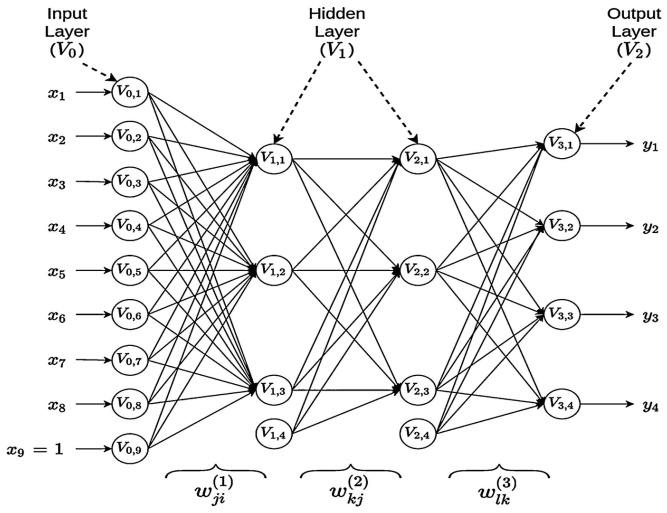


Figure 2.18: Schematic of a Deep Neural Network, obtained from [65]

Graph neural network interatomic potentials (GNN-IPs) have risen as very powerful MLIP networks for deep learning of interatomic potentials, which eliminate the need for hand-crafted descriptors, and instead can learn graph representations of atoms from invariant features of geometric data. Here, atomic structures are represented by collections of nodes and edges, where nodes in the graphs correspond to individual atoms and edges, and these are defined by connecting every atom to all other atoms that are closer than a defined cutoff distance. Examples of these GNN-IPs include NequIP [66], TeaNet [67], and MACE [68]. These are also known as the E(3) equivariant neural networks, which can work with highly expressive equivariant tensor representations of atomic environments, and operate on them to preserve the proper symmetries [63].

MLIPs are generally trained on a limited amount of chemical or materials systems' information, due to there being a specific application or system of interest for which the method is developed. Therefore, these models sometimes suffer from transferability issues in which one model works very well for a specific application or material, but might not show good fitting results for other systems, due to the inability of extrapolating accurately modeling new elements or structures which are not present in the training data used for the creation of the model [63].

To address the scalability issues seen in AEFs-MLIPs, universal MLIPs (uMLIPs) are trained with the largest consistent datasets available. These algorithms are able to produce a potential with the widest possible domain of applicability, where it is then possible to research the dynamics of a diverse number of chemically complex systems. These attempt to cover a large number of species under different levels of constraints. They can cover between 10-100 elements in different conditions and configurations. Table 2.2 shows a summary of different U-MLIPs that have been recently developed (and most are in continuous development). These U-MLIPs are all based on previously developed GNN models to include physical information of how bond energies of systems evolve with the change in position of atoms (due to a MD simulation taking place). Forces and stresses are then acquired via differentiation of this learned energy dependence. These models have made possible the growth of large computational databases containing millions of DFT calculations. Each DFT structure provides one energy and 3N forces (N being equal to the number of atoms in the structure), which are then used as training data for the universal MLIP [63, 69].

Multiple benchmarking studies [81–85] revealing the performance of U-MLIPs have been recently published. Here, different MLIP models are used for various material properties. In general, three or more U-MLIPs are chosen to study properties such as cell relaxations, bulk and surface total energy calculations, material stability (where materials with a convex hull energy within some threshold are found via structural optimization and active learning), and mixing energy of binary alloys, just to name some examples. The performance of the selected models is then compared, where parity plots comparing the MLP calculated property (such as energy or forces) and the DFT calculated property (also known as the "ground truth") are made. A measure of the error between the DFT and MLIPs values is also calculated, usually in the form of the mean average error (MAE) or root mean square error (RMSE). These are defined in equations 2.14 and 2.15, respectively.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - x| \tag{2.14}$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{(y(i) - \hat{y}(i))^2}{n}} \tag{2.15}$$

Accurate performance of these MLIPs has been reported in these studies. For example, figure 2.19 shows an example of the U-MLIPs' performance on the calculated surface, defect, and solid solution

Table 2.2: Overview of Universal Machine-Learned Interatomic Potentials (U-MLIPs), adapted from [63]

Model Name	Description	Elements Covered	Training Database
PFP (Preferred Potential) [70]	Tensorial message passing neural network, now a commercial product in Matlantis.	96	Over 59 million diverse structures (evolved from 10M in v1 to 59M in v7)
GNoME (Graph Networks for Materials Exploration) [71]	A custom-trained version of NequIP, fit to an in-house database.	94	In-house database of roughly 80 million DFT calculations
MatterSim [72]	Large-scale deep learning model trained on actively-learned DFT data, covering a wide range of temperatures and pressures.	118	Large custom database of roughly 17 million atomic configurations, including non-ground state structures
GPTFF (graph-based pre-trained transformer force field) [73]	A GNN model with transformer blocks integrated into the model architecture.	N/a (related NEP89 covers 89 elements)	OMat24 (over 110 million structures for inorganic bulk materials) and other datasets (MPtrj, SPICE, ANI-1xnr)
EquiformerV2-OMAT24 [74]	Trains the EquiformerV2 model on a novel open-source database, showing best performance on MatBench leaderboard (as of writing).	88 (OMat24 dataset)	OMat24 (roughly 118 million atomic configurations)
Orb [75]	Achieves excellent performance on MatBench leaderboard, offering faster performance (especially for large systems).	89	Fine-tuned version of an internal foundation model called LINUS
MACE-MP0 [76]	Equivariant graph tensor network demonstrated for accuracy on various applications and stable MD simulations.	89	MPtrj (same publicly available data as CHGNet and SevenNet)
SevenNet-0 [77]	Based on NequIP, refined for good scaling on many processors for larger systems.	89	Materials Project data (same as M3GNet)
M3GNet (Materials Graph NETwork) [78]	A 3-body potential aiming for broad applicability.	89	Materials Project data
CHGNet (Crystal Hamiltonian Graph Neural Network) [79]	Graph neural network considering atomic magnetic moments.	89	MPtrj (Materials Project relaxation trajectories, 1.58M structures)
ALIGNN-FF (unified atomistic line graph neural network-based force field) [80]	Unified atomistic line graph neural network-based force field for structurally and chemically diverse solids.	89	JARVIS-DFT (75,000 materials, 4 million energy-force entries, with a subset of 307,113 used for training)

energies of various surfaces composed of varying chemical systems. Figure 2.19a shows the parity plots of DFT surface energies versus the MLIP calculated surface energy from three different U-MLIPs, M3GNet, CHGNet, and MACE, and figure 2.19b shows the same but with the defect energies. 147 surfaces with multiple Miller indices of 29 elements and binary compounds were evaluated for the surface energies; meanwhile, for the defect energies, 129 point defects across 32 chemical systems are included. All three models used show very close MAE, with MACE having the lowest error of them all. Nevertheless, the three models underestimate the surface and point defect energies compared to DFT; however, this is due to a universal softening of the PES, which is then resolved by fine-tuning/retraining these models via fine-tuning the model with high-energy out-of-distribution training points [63, 83].

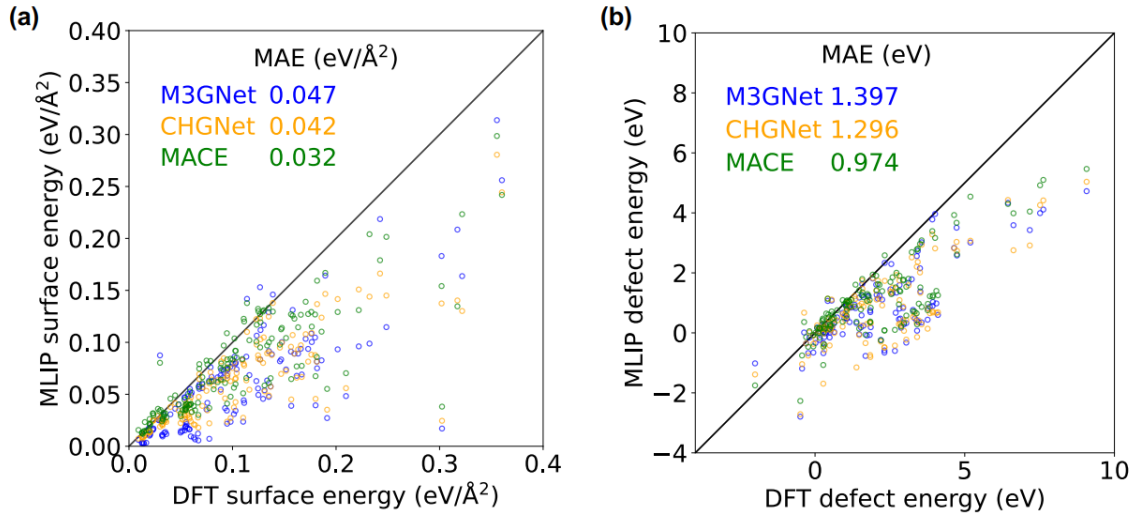


Figure 2.19: U-MLIP performance on surfaces and defects, obtained from [83]. a) Parity Plot of DFT surface energies and MLIP surface energies. b) Parity plot of DFT defect energies and MLIP defect energies.

2.5. MACE

MLIPs have shown great potential as candidates to substitute the use of DFT, especially when they are coupled with databases containing millions of material properties that help refine the model. These MLIPs are usually made available through GitHub repositories, where it is possible to download and use the models to train, evaluate, and produce force fields. Examples and tutorials are also given to give a deeper insight into the model's working mechanism [68, 72, 75, 76, 86].

One major feature that was also shown in the previous section is that to increase the accuracy of these models, fine-tuning or retraining of the model using data that does not meet certain thresholds, for example, a high energy error, is required. This fine-tuning is game-changing in the case of modeling (something about catalysis). However, even though these models do offer fine-tuning Python scripts, the user is forced to manually access them and modify the input files for their usage. And this, of course, doesn't include all the previous data cleaning, model training, and post-validation of the model via the implementation of, for example, MD simulations [63, 83].

Therefore, one of the major objectives of this thesis project is the engineering of an active learning loop workflow for the universal usage of a machine learning interatomic potential algorithm. In this case, MACE is used to generate novel machine learning force fields for a Ni cluster catalyst supported on Ga_2O_3. MACE was chosen as the MLIP algorithm to work with due to its ease of use and access to pre-trained foundational models, which are trained on datasets with different levels of theory (PBE+U, PBE, r2SCAN). Additional information about MACE and its usage may be found in the GitHub ACE-

suit repository [68]. MACE possesses a novel MLIP architecture, where a combination of equivariant message passing with efficient many-body messages is established. As previously mentioned, it is capable of achieving state-of-the-art performance on different benchmarking tests, while also being able to display greater generalization capabilities over other approaches on extrapolation simulations [68, 87].

MACE is an equivariant graph neural network. More specifically, it is known as a message passing neural network (MPNN). MPNNs are a subtype of GNNs that parametrize a mapping from a labeled graph to a target space, which can be in the form of a graph or vector space. Applying this parametrization to the properties, materials, or molecules embeds the graph in a 3-dimensional Euclidean space. In this space, every node represents an atom, and edges connect the nodes only if the corresponding atoms are located at a certain distance between each other [68].

2.5.1. Theory and Mathematics

In MPNNs, multiple message constructions, updates, and readout steps are made to read out and learn the state of a node at a specific layer of the neural network. Specific information about how each of these steps is mathematically represented may be found in Appendix B. MACE's architecture follows this message-passing structure; however, there is a key difference in the message construction mechanism. As shown in equation 2.16, the messages are expanded in a hierarchical body order expansion, where a hyper-parameter ν corresponding to the maximum correlation order (which is the body order minus 1) of the message function concerning the states is applied. Multiple learnable functions ($u_1, u_2 ... u_\nu$) are used, and the sums iterate over the neighbors of atom i. The overall body order of the positions may be greater, since this depends on the body order of the states themselves. Most importantly, the sum over $j_1, ... j_\nu$ incorporates self-interaction and yields a tensor product structure which is the key towards computationally efficient parametrization. This evades the exponential scaling of computational cost with the correlation order ν. This way, using higher body order messages are able to make multiple relationships with different numbers of layers and expressivity of the network, which results in a fast, highly parallelizable model [68].

$$\mathbf{m}_i^{(t)} = \sum_j \boldsymbol{u}_1(\sigma_i^{(t)}; \sigma_j^{(t)}) + \sum_{j_1, j_2} \boldsymbol{u}_1(\sigma_i^{(t)}; \sigma_{j1}^{(t)}, \sigma_{j2}^{(t)}) + ... + \sum_{j_1 ... j_\nu} \boldsymbol{u}_1(\sigma_i^{(t)}; \sigma_{j1}^{(t)}, ..., \sigma_{j_\nu}^{(t)}) \tag{2.16}$$

For a better grasp of the underlying theory and foundations of MACE, it is important to understand the meaning behind equivariance and its role in graph neural networks. Generally, geometric passing GNNs propagate only local scalar quantities like angles and distances. Equivariant GNNs, on the other hand, can propagate geometric quantities like vectors and high-order tensors. High-order spherical tensors are an essential part of the MACE architecture [88], and also give it the label of an e3nn (also known as Euclidean neural networks). These networks operate on geometry and geometric tensors which describe systems in 3D, and can transform predictably under a change in the coordinate system [89]. The working mechanism of these tensors is reflected in equation 2.17, where $\tilde{\mathbf{h}}_{i,l} \in \mathbb{R}^{(2l+1)\times f}$, begin at an order of $l = 0$ for scalar quantities, 1 for vector quantities, and go as far as arbitrary orders of $l = L$. These tensors are then updated via tensor products $\bigotimes$ of neighborhood features $\tilde{h}_j$ and the spherical harmonic representation Y evaluated on the unit vector of the relative positions ($\frac{\vec{x}i,j}{|\vec{x}i,j|} = \hat{x}ij$) [88].

$$\tilde{\mathbf{h}}_i^{(t+1)} := \tilde{\mathbf{h}}_i^{(t)} + \sum_{j \in \mathcal{N}_i} Y(\hat{x}_{ij}) \otimes_{\mathbf{w}} \tilde{\mathbf{h}}_j^{(t)} \tag{2.17}$$

MACE's architecture is able to address some of the major limitations seen in previously developed MLIPs. For example, the atomic cluster expansion (ACE) provided a framework for developing high-

body-order complete polynomial features independent of the body order. However, these models are limited by their cutoff distance and relatively rigid architecture compared to MPNNs, which leads to less accurate MLFFs. Coupling MACE with ACE provides a route for efficient computation of high k-body order features in the Euclidean neural network framework. Due to this coupling, symmetrisation or generation of all k-tuples in more standard many-body expansions are evaded [68, 88]. Equivariant MPNNS, such as NequIP, suffer from high computational cost due to using L=3 spherical tensors as messages and 4 to 6 passing iterations. This also led to difficulties in parallelization across multiple GPUs [76].

2.5.2. MACE Foundational Model and Active Learning

As previously mentioned, another advantage of MACE usage is its readily available foundational models. These are pre-trained models using databases with DFT data. An example is MACE-MP0, which is trained on the Materials Project dataset on 89 different elements on a PBE+U level, as it was described in table 2.2. This model has been able to tackle different tasks and predictions, such as running stable MD simulations over different chemical systems, predicting phonon spectra, calculating activation energies for defects, simulating solvent mixtures, and modeling hydrogen combustion [76]. Figure 2.20 shows a diagram of the elements included in the MP0 training dataset, where different configurations, compositions, morphologies, and properties of these material systems are included [76].

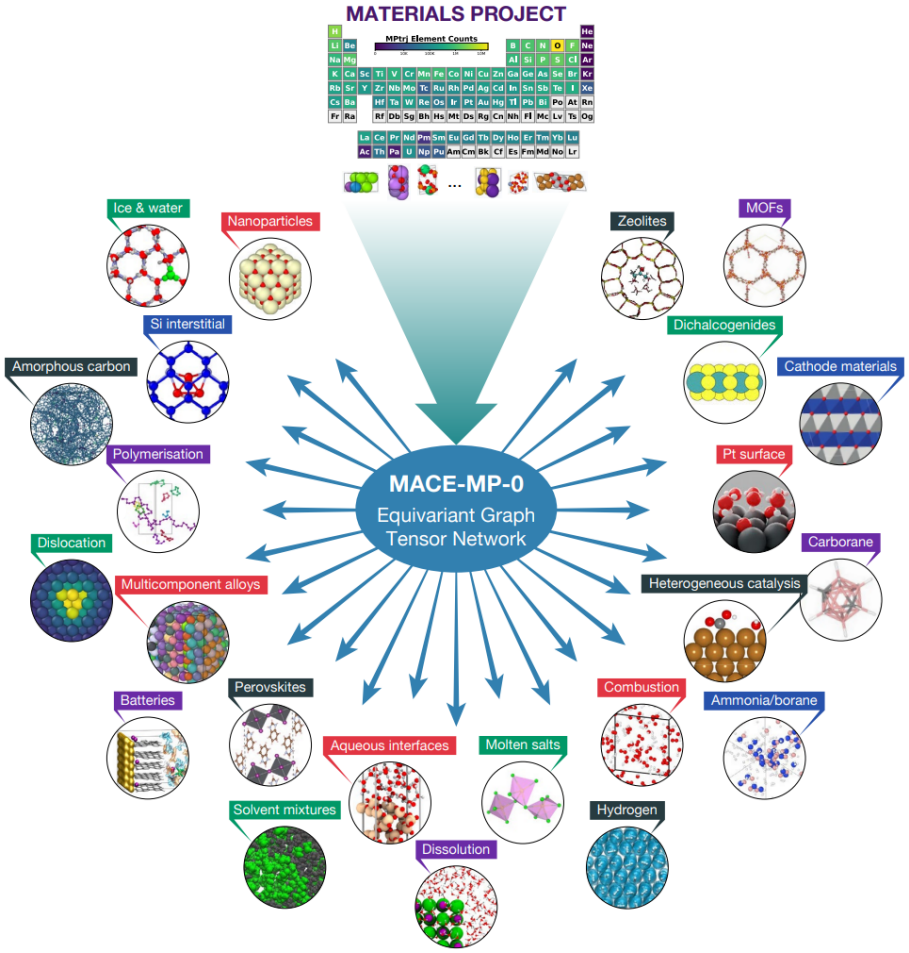


Figure 2.20: MACE-MP0 as a foundational model for materials modeling. Trained on Materials Project Data, with the capacity of powering MD simulations [76].

In catalysis, for example, this foundational model has been used as a substitute for DFT in the calculation of reaction barriers and rates of several heterogeneous catalysis reactions. Reportedly, MACE-MP0 has shown a great agreement with DFT based calculations on multiple applications, such as in the construction of Porubaix diagrams for bulk CuO and Pt(11) surface; the calculation of adsorption energy scaling between O and OH on different transition metal surfaces; and finally, the reaction energy profiles for CO oxidation on Cu, as well as a key step in the conversion of CO_2 to methanol on an In_2O_3 surface [76]. It has also shown the capacity of matching DFT predictions on zeolites and silica polymorphs to study their phase transitions at high pressures [90]. As a final example, MACE-MP0 was used as a potential for the calculation of rate-determining steps to further understand the mechanism of oxygen reduction reaction (ORR) overpotentials on Pt-based catalysts [91]. Therefore, MACE has been taken as the choice for catalyst research in this project.

The foundational model is, however, not always quantitatively accurate for most catalysis applications, yet its reported stability for MD simulations and exploring reactive pathways makes it a great starting point for catalytic dynamics calculations. To further improve these models, active learning and/or fine-tuning are recommended using relevant configurations or phase space regions. After doing first point calculations with this data, these could be introduced as fine-tuning datasets to construct a more robust, application-focused potential to obtain more accurate energetic and forces calculations [63, 76, 83].

An example of an active learning loop is shown in Figure 2.21. The starting training set is used to train an initial MACE model. This model can be a completely new MACE model or a finetuned foundational model (MACE-MP0). Afterwards, the successfully trained model is used to run MD simulations, and then a comparison must be made between the ground truth MD run versus the MACE-based MD simulations. To make this an automated process, error predictions may be placed, and according to a set threshold, certain configurations are selected and inserted into the original training set. The model is then finetuned, and the process begins anew until the error is lower than the specified threshold [68, 76].

Comparing the radial distribution functions of both the ground truth, DFT-based trajectory, versus the MACE-based MD trajectory is another method to prove if the MLIP model provides accurate force fields that mimic precisely the DFT-based potentials [68, 76].

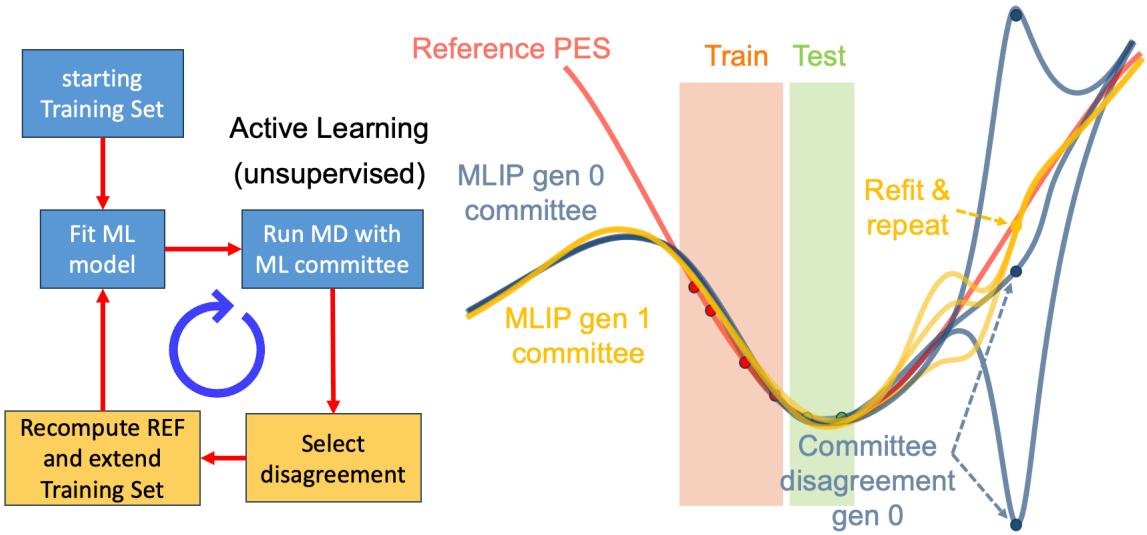


Figure 2.21: Active Learning Example, obtained from [68, 87]

3

Methodology and Research Outline

In this project, molecular dynamics (MD) simulations will be performed on a nickel-supported gallium oxide structure (Ni/Ga_2O_3) to analyze its thermal stability and calculate energetic barriers. Ni/Ga_2O_3 has emerged as a promising novel catalyst for the CO_2 reduction reaction, demonstrating high selectivity for CO_2 hydrogenation to methanol [14, 92–95]. It has also been shown to be effective in the reverse-water-gas-shift (RWGS) reaction, where CO_2 is converted into carbon monoxide and water when reacted with H_2 [14].

3.1. Initial Data Processing

A large set of Ni/Ga_2O_3 configurations, provided by Margareth Baidun from the Inorganic Systems Engineering group at TU Delft, will be used for this study. These include 464 relaxed configurations of Ni/Ga_2O_3, as well as over 1000 additional configurations with a pure Ga_2O_3 support or with varying concentrations of hydrogen atoms. To streamline the project and establish a proof of concept, this research will primarily focus on simulations involving the Ni/Ga_2O_3 catalyst without the presence of hydrogen atoms.

Performing Density Functional Theory (DFT) simulations for all of these relaxed configurations would be computationally prohibitive due to the large size of the files and the complexity of the structures, which contain, on average,168 atoms. A typical configuration, as shown in Figure 3.1, features a cluster of eight nickel atoms on top of a Ga_2O_3 support.

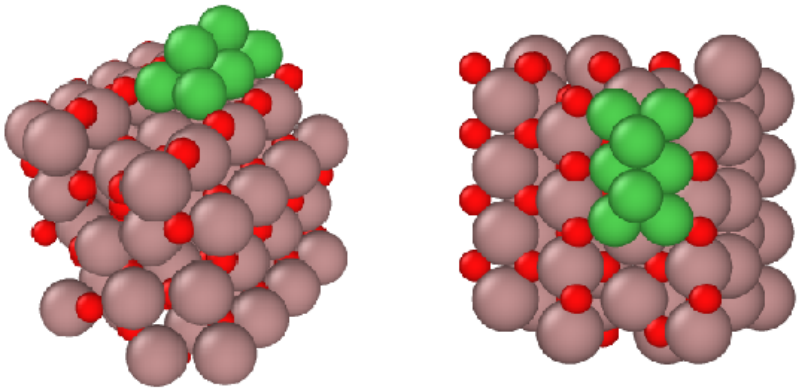


Figure 3.1: Side and Top views of Ni/Ga_2O_3 catalyst

Therefore, this thesis aims to demonstrate the viability of using both self-trained machine learning models and foundational models (MP0) to perform accurate and comparable MD simulations of these structures. This approach allows for a detailed analysis of the thermal stability of the catalyst over a determined number of time steps. This project will address the following research questions:

1. To what extent can a MACE potential serve as an adequate substitute for traditional DFT-MD simulations for the Ni/Ga_2O_3 system?
2. Do foundational machine-learning models exhibit inherently lower errors in predicted energies compared to in-house trained models for the Ni/Ga_2O_3 system?
3. Is the implementation of an active learning loop sufficient to achieve better overall performance in MD simulations of the Ni/Ga_2O_3 catalyst?
4. How does the calculated energy barrier for the formation of the formate ion on the Ni/Ga_2O_3 catalyst using the MACE-MP0 foundational model compare to that obtained from traditional DFT potentials?

3.2. MACE Usage and Initialization

Firstly, it is important to mention that most of the simulations that will be presented here were done locally using Jupyter notebooks. For the successful usage of MACE and running simulations with it, the packages seen in table 3.1 were installed in a custom-made conda environment. Most of the other required packages will be installed simultaneously with the installation of other packages due to their dependencies on each other. Something important to note is that much debugging and package installation with other methods, such as conda forge, was required for certain packages, as well as some packages only being available in Linux or Python versions.

Table 3.1: Table of relevant Python packages for MACE usage and their documentation

Package Name	Description	Documentation
ASE	The Atomic Simulation Environment is a powerful set of tools and Python modules for setting up, running, and analyzing atomic simulations.	ASE Wiki
cuequivariance-torch-cu	This is a specific library variant providing CUDA-accelerated equivariant neural network operations, typically used in conjunction with PyTorch for advanced geometric deep learning tasks. Can be used with CUDA version 11 or 12. Only available in Linux.	PyPi package
mace_torch	MACE (Machine learning interatomic potentials) is a fast, high-accuracy framework for interatomic potentials, built on PyTorch for efficient MD simulations.	MACE GitHub
Matplotlib	A comprehensive library for creating static, animated, and interactive visualizations in Python. It includes the widely used `pyplot` and `pylab` modules for plotting.	Matplotlib Docs
Numpy	The fundamental package for scientific computing with Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of high-level mathematical functions.	Numpy Docs
pandas	A powerful library for data manipulation and analysis. It provides flexible data structures like DataFrames, which make working with structured data intuitive and efficient.	pandas Docs
PyTorch	An open-source machine learning framework that accelerates the path from research prototyping to production deployment. It provides powerful tensor computation with GPU acceleration.	PyTorch Docs
SciPy	A library of algorithms and mathematical tools for scientific computing, built on the NumPy extension.	SciPy Docs
torchaudio	A PyTorch-specific library for audio signal processing and machine learning, providing datasets, transformations, and pre-trained models.	torchaudio Docs
torchvision	A PyTorch-specific library for computer vision, providing access to popular datasets, model architectures, and common image transformations.	torchvision Docs

Furthermore, to record all of the relevant Python scripts and notebooks developed for this project, a GitHub repository is available to access all of this data, which is available **here**.

3.2.1. VASP Data and Conversion

The very first runs with MACE were done with in-house trained models. For this, training data and testing data for the models had to be recollected and processed for MACE to read them successfully. MACE can read .xyz-formatted files as its input for training, testing, and validation datasets. For this

application and considering that the data that was given is in Vienna Ab Initio (VASP) simulation output folders (where OUTCAR, CONTCAR, VASPRUN, and other files are found), it is essential to create a script where the configurations found in these files are read and placed in an xyz file. In the github, the vaspexp.py script can be directly used to write an XYZ file containing the data from one or multiple vasprun.xml files. The only input parameter required is the folder path for the different vasprun output folders, and then multiple xyz files will be outputted. One of them is an xyz file including all of the configurations in a single file, and the other xyz file containing only the configurations of a single vasprun file.

As previously mentioned, VASP was used to perform structural relaxations. This is to allow atoms to move to their lowest energy positions. To perform these experiments, VASP version 5.4.4 was used through the HPC cluster of the Mechanical Engineering Faculty in TU Delft. To perform these relaxations, four files must be configured correctly. These are the INCAR, KPOINTS, POSCAR, and POTCAR files. The INCAR file includes all of the initialization for the VASP calculation, where the type of calculation (MD or DFT) is defined, as well as convergence criteria and other parameters. KPOINTS defines the k-point mesh used to sample the Brillouin zone during calculations. In the POSCAR file, the initial configuration for your system is defined, where the position of all the atoms from every single element is defined. And finally, the POTCAR file, which utilizes projector-augmented-wave (PAW) pseudopotentials, determines the level of exchange-correlation functional used for the calculations, such as the PBE functional. Additional information on the configuration for the INCAR file may be found in appendix C, where notably the ”NSW” parameter has to be changed from 300 to 1 for single point calculations. More information on VASP and how to utilize it may be found on the VASP documentation [41].

After VASP has run and converged, the CONTCAR file contains the final relaxed structure. Other files, such as VASPRUN and OUTCAR, contain all the information regarding all the configurations per step, where forces, energy, and atomic positions are also recorded. These files are then read by the vaspexp.py script, and then the respective xyz files are made.

3.2.2. MACE Training and Evaluation

After processing the VASP outputted configurations, the following task is to set up the MACE training. For this, a training, testing, and validation set must be defined. Typically, an 80/20 train/test setup is utilized when training an ML model. And in the case of the validation set, MACE has a command to take a user-defined percentage of the training set as the validation set. Two Python scripts were developed for MACE training, one for training a single model and another for training a committee of models (MACE_training.py and MACE_committee_training.py, respectively). A committee of models refers to a group of trained models over the same dataset; however, the training data is completely randomized for each of the models, making them similar but not entirely the same. The final prediction of the committee model is the average of all committee members when using it, for example, as a calculator in the MD simulations. This approach has been shown to decrease errors in predictions [96].

Both of these codes work in a very similar fashion, where, as input, the previously produced xyz file with the combined configurations (or an individual xyz file with the configurations of one single VASP run if the user would prefer) is taken. The configurations are then read and placed in a list using the ASE.IO package, randomized to produce a completely mixed, unbiased dataset, and then split into the training and testing datasets. These are then saved in a file so that the user would like to access them afterwards.

The second part of the script is the MACE training itself. An example of how the training setup and commands used are seen in listing 3.1. Here, the method begins by setting up logging infrastructure,

creating a log file in the output directory that will capture all training output and errors. This is crucial for monitoring training progress and debugging any issues that may arise during the potentially long training process. The core functionality revolves around constructing a comprehensive command-line argument list for the mace_run_train executable, which is MACE's primary training Python script.

```python
def train_model(self):
    """Executes MACE model training using the specified dataset."""
    log_file = os.path.join(self.output_folder, f"{self.model_name}_training.log")
    command = [
        "mace_run_train",
        f"--name={self.model_name}",
        f"--train_file={self.train_file}",
        "--valid_fraction=0.05",
        "--forces_key=forces",
        "--energy_key=energy",
        f"--test_file={self.test_file}",
        "--config_type_weights={'Default':1.0}",
        "--E0s=average",
        "--model=MACE",
        "--hidden_irreps=64x0e + 64x1o",
        "--r_max=5.0",
        "--batch_size=5",
        "--max_num_epochs=100",
        "--stage_two",
        "--start_stage_two=50",
        "--ema",
        "--ema_decay=0.99",
        "--amsgrad",
        "--restart_latest",
        f"--device={self.device}",
        "--swa"
    ]
    # Add --enable_cueq argument based on input
    command.append(f"--enable_cueq={str(self.enable_cueq)}")
    with open(log_file, "w") as log:
        subprocess.run(command, stdout=log, stderr=log, text=True)

```

Listing 3.1: Example of the Python function to train a MACE model.

The model uses a validation fraction of 5% (–valid_fraction=0.05), which is a reasonable split for monitoring overfitting during training. The architecture is specified through hidden_irreps=64x0e + 64x1o, which defines the irreducible representations used in the equivariant neural network layers - this notation indicates 64 scalar (0e) and 64 vector (1o) features, ensuring the model respects rotational and translational symmetries of atomic systems. In the MACE GitHub, it is set to 128x0e + 128x0e, but this proved to be too computationally heavy for running models locally, and an accurate performance of the model was still expected to be seen with a lower number of scalar and vector features [68, 87]. However, the user is free to change this to higher or lower values to determine if better model performance is attained.

The training strategy employs a two-stage approach with stage_two and start_stage_two=50, where the model likely transitions from a simpler to a more complex training regime at epoch 50 out of 100 total epochs. This is complemented by several advanced optimization techniques: Exponential Moving Average (EMA) with a decay rate of 0.99 for model parameter smoothing, AMSGrad optimizer for improved convergence, and Stochastic Weight Averaging (SWA) for better generalization. The inclusion of restart_latest ensures training can resume from checkpoints if interrupted [68, 87].

The physical parameters are set appropriately for atomic systems, with an interaction cutoff radius of 5.0 Ångströms (r_max=5.0) and atomic reference energies calculated as averages (E0s=average). These atomic reference energies are the energies of the isolated atom itself. In this case, the average was chosen since MACE estimates the atomic energies using least squares regression. This could result in less stable potentials, so it is recommended that if the user knows these values, they insert them manually for each atom, as it is shown in the MACE repository [68, 87]. The small batch size of 5 is typical for MD datasets, where individual configurations can be computationally expensive to process. Finally, the method dynamically adds the enable_cueq parameter based on instance configuration (it is also set as an input parameter when calling the script, since this parameter requires the cuequivariance_torch_ops_cu); however, I did not have access to it in one of the clusters utilized for testing. The entire training process is executed while redirecting all output to the log file for comprehensive monitoring and debugging capabilities.

After training has finished, MACE automatically evaluates the model and creates two parity plots, where comparisons between the energies per atom and forces of the training, testing, and validation models are made. MACE also shows the model training performance, showing the training and validation loss over the number of epochs defined for training. It also shows a graph right next to it where the RMSEs of energies and forces are plotted over time. An example of these results is shown in Figure 3.2, showing the performance of one of the first successfully trained models. This model was trained with only one VASP output folder, containing a total of 98 configurations. No over- or underfitting of the model is observed either, which implies that the input parameters stated previously (batch size, number of epochs, and hidden layers, for example) are correctly optimized.

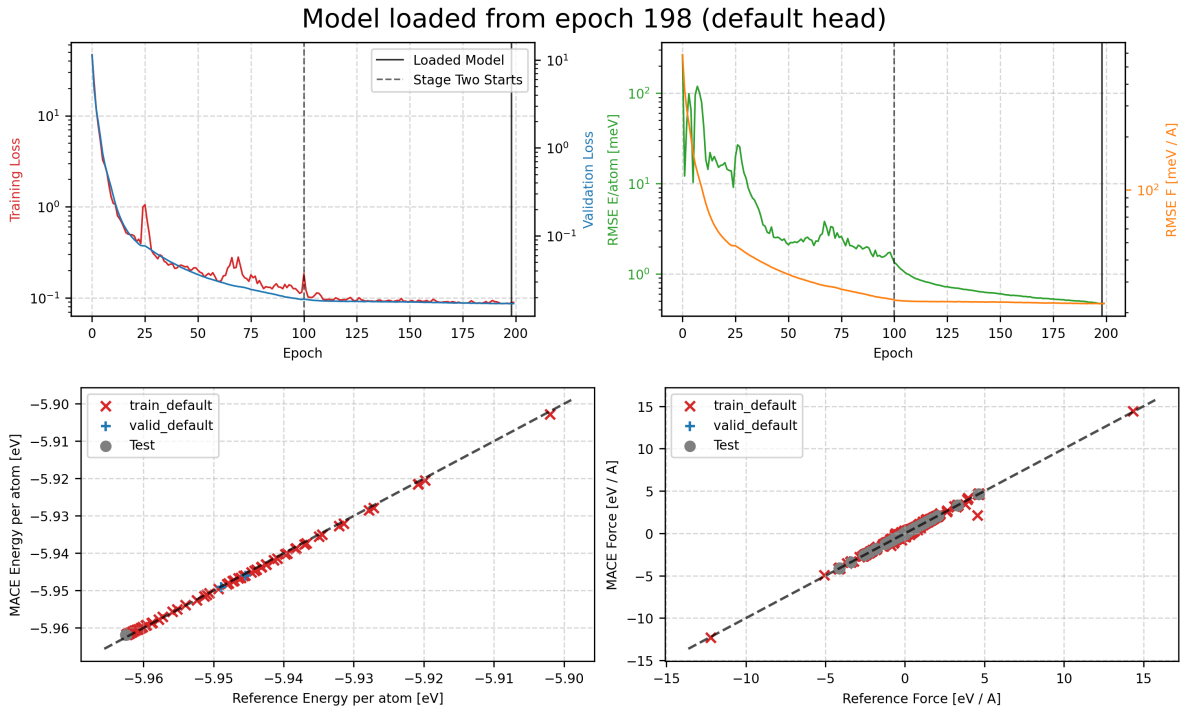
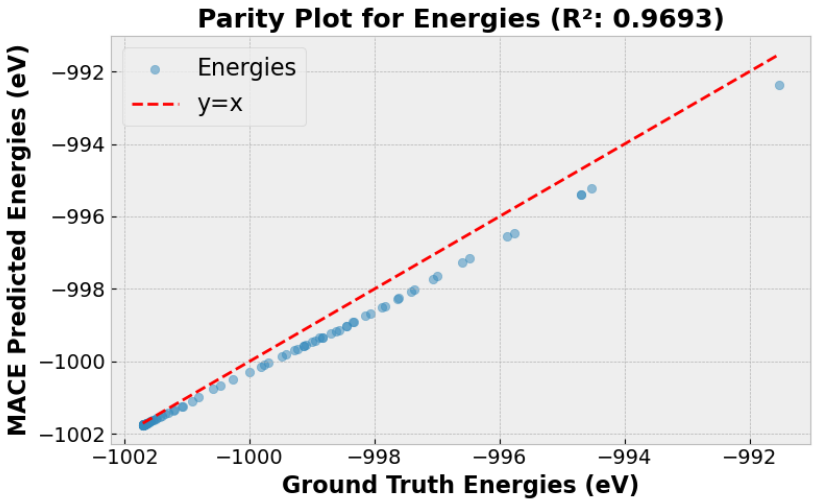


Figure 3.2: Initial MACE Training Results

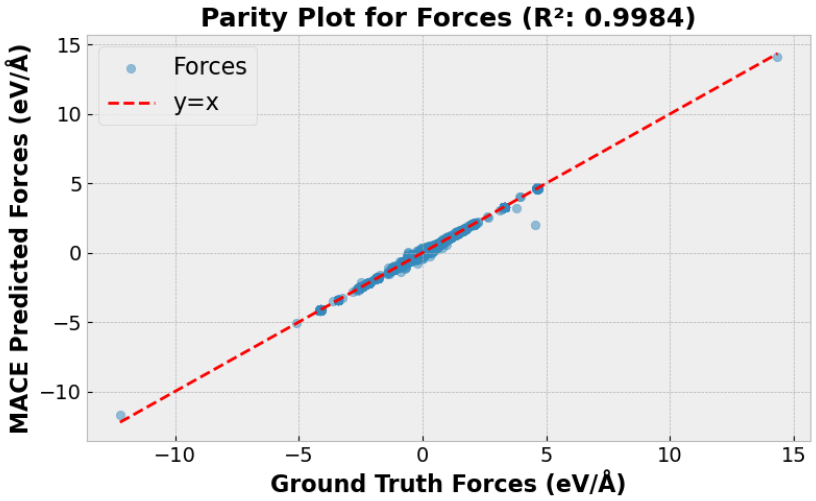
Further evaluation of the model may be done by using mace_eval_configs.py script provided by the MACE package. This script evaluates configurations contained in an XYZ file with a trained model. This script was initially used to compare the initial configurations used for training and testing (so the combined xyz file outputted from vaspexp.py would be used as an input), and it outputs an xyz file

containing the configurations outputted after the trained model evaluates them. This way, it is possible to do parity plots manually of the energy per atom and forces to check the accuracy of the model, where the R^2 for each property is also calculated between the "ground truth, DFT data" versus the predicted MACE data. A script was also made for this evaluation named "committee_eval.py" Examples of two parity plots done for the model mentioned previously are shown in Figure 3.3. A pretty good fit between both datasets is observed, both with an R^2 higher than 0.96, therefore showing that the model training was done successfully, and it provides an accurate model.

Figure 3.3: Parity Plots of Energies and Forces for Initial MACE Training



(a) Energy Parity Plot



(b) Force Parity Plot

MLIPs already prove to be a good starting point for substituting DFT-based calculations, as it has been mentioned in previous sections. However, one essential part of constructing these models is the validation step. For this, different simulations can be engineered to validate the potentials. In this research, MD using a canonical ensemble will be used with Langevin Dynamics. In a canonical ensemble, the number of atoms (N), system temperature (T), and volume (V) are fixed (which is why this ensemble is also referred to as an NVT ensemble). This way, it is possible to run dynamics to describe systems that

are in thermal equilibrium and can exchange energy with their environment. Usually, DFT is used using software such as VASP or LAMMPS coupled with pre-calculated pseudopotentials (like the aforementioned PAW potential files). However, using the atomic simulation environment (ASE), it is possible to use MACE models directly as potentials or ”calculators” for the MD simulations. Comparisons may then be made between the performance of an MACE model with the ”ground-truth” DFT MD run.

3.2.3. Molecular Dynamics with ASE

To run MD using ASE, two Python scripts named MACE_MD.py and Simple_MD_Found.py were developed. The former is used for MD simulations with the self-trained MACE potentials; meanwhile, the latter takes the MACE foundational model (MP0) as its calculator. The first script takes as input an xyz file (which can be the same combined xyz file used for the training), the temperature at which the MD simulation would take place, the number of steps to run the simulation, and the paths of the trained MACE model(s). The second takes the same inputs except for the model paths, where instead the user is able to define what foundational model they would like to use (MP0 small, medium, or large). For more information and other available foundational models, it is recommended to refer to the list shown in the MACE GitHub [68, 87]. It is also possible to define a z-threshold, since inside this script, a constraint is placed where the substrate atoms are set to be stationary. This is a common practice in surface chemistry simulations, since it allows adsorbates and surface layers to move freely.

The simulation setup employs sophisticated MD protocols, initializing velocities according to a Maxwell-Boltzmann distribution at 300K before switching to the user-specified target temperature for the actual simulation. Whereby default, it is set to 323.15K. The class uses a Langevin thermostat with a 1 femtosecond timestep and 0.1 friction coefficient, providing good temperature control while maintaining realistic dynamics. While the simulation is running, it is possible to observe a real-time visualization of the temperature and energy fluctuations over each timestep, which provides immediate feedback on simulation quality and convergence. If using a committee of models in MACE_MD.py, a graph showing the variance in energy for all of the models is shown. This can serve as important information to see how similar the models are at performing MD, and to point out where the largest deviations are located throughout the simulation. Figure 3.4 shows a comparison between a well-converged MD run versus an MD run that did not converge very well due to instability of the material. The unstable simulation was done at a very high temperature (1200K), with a potential that was not trained with H atoms (since this was a test using a Ni/Ga_2O_3 structure that did contain hydrogen in its surface). This could have been due to either not having setup a correct z-threshold, the temperature of simulation was too high for this material (the simulation was done at 1200K), or the potential is not well trained (due to low amount of configurations used for training, or there were missing elements, like hydrogen, in the training of these potentials).

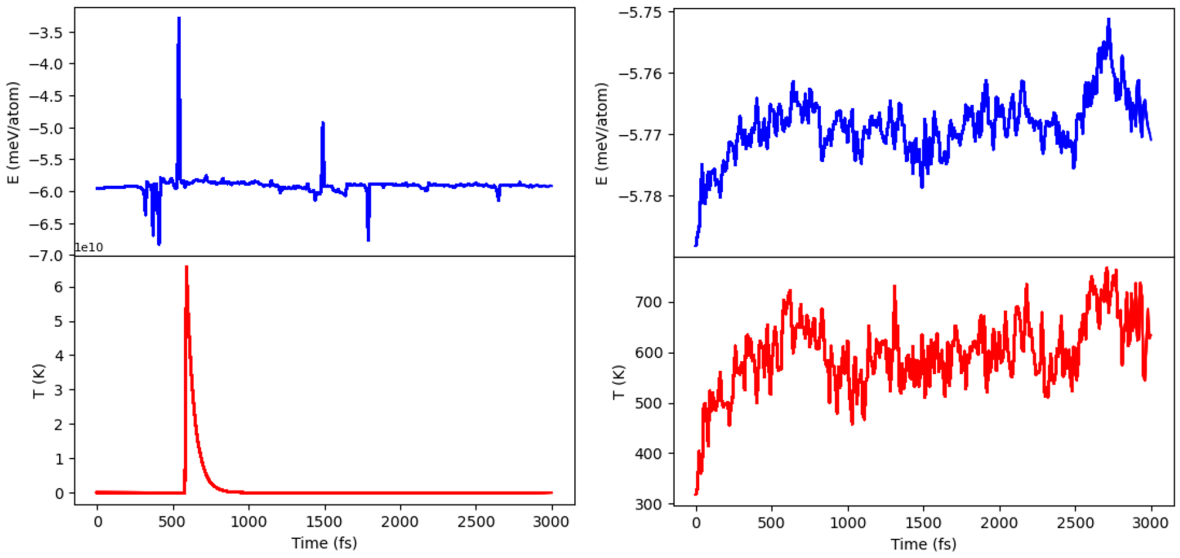


Figure 3.4: Left: Unstable MD simulation, leads to explosions in the trajectory. Right: Converged MD simulation using MACE-MP0 as calculator

As outputs of these scripts, a log file containing the kinetic, potential, total energy, and temperature per timestep is saved; a trajectory file and xyz file containing all the configurations per timestep, as well as their energies and forces are saved; and lastly, an image of the total energy and temperature versus time is saved.

3.2.4. Molecular Dynamic Analysis

For analysis of the outputted trajectory files, OVITO [97] was used to visualize all of the configurations over time. With OVITO, it is possible to observe the movement of atoms in the system cell. An example of a configuration visualized via OVITO is shown in Figure 3.1.

To plot the energies of the MD simulation, a Python script named energies_plot.py was developed. In this script, a trajectory file from the MD simulations is placed as an input. This script outputs three plots, one for the kinetic, potential, and total energy versus time, respectively. This also includes a visual representation of the mean value per energy and its respective standard deviation. These values would also be printed out once the script is run.

A Python script named avg_coordination.py was developed to have a deep coordination environment analysis over the last configuration of a given xyz file. There are multiple output files and images outputted by this code. Currently, it is configured specifically for the Ni/Ga_2O_3 system, since there are some lines of code where only the Ni bonds are read. For example, in the coordination number distribution of the Ni-Ga bonds, the frequency of coordination numbers is shown over the entire structure. Nearest neighbor distributions of all bond types present in the system (so Ni-Ni, Ni-Ga, Ni-O, for example) are also recorded and plotted versus their frequency of appearance. Lastly, a final modification of the system was done to study the generalized coordination number (GCN) of the Ni cluster on the surface of the catalyst. The generalized coordination number is a proficient and novel descriptor for the activity of nanocatalysts [98, 99]. Therefore, it was of interest to see if these clusters formed highly coordinated environments to have an initial guess at the structure-activity relationship.

3.3. Active Loop Engineering

To possibly achieve more accurate machine learning based MD simulations, an active learning loop was developed to decrease the error in predicted energy. As previously mentioned in section 2, finetuning or retraining the models with outlying data shows an increase in model performance. A schematic depicting the different steps of the loop is shown in Figure 3.5. Here, all of the aforementioned steps are included, plus some additional ones, where VASP is used to do single-point calculations of selected configurations from the MD simulation.

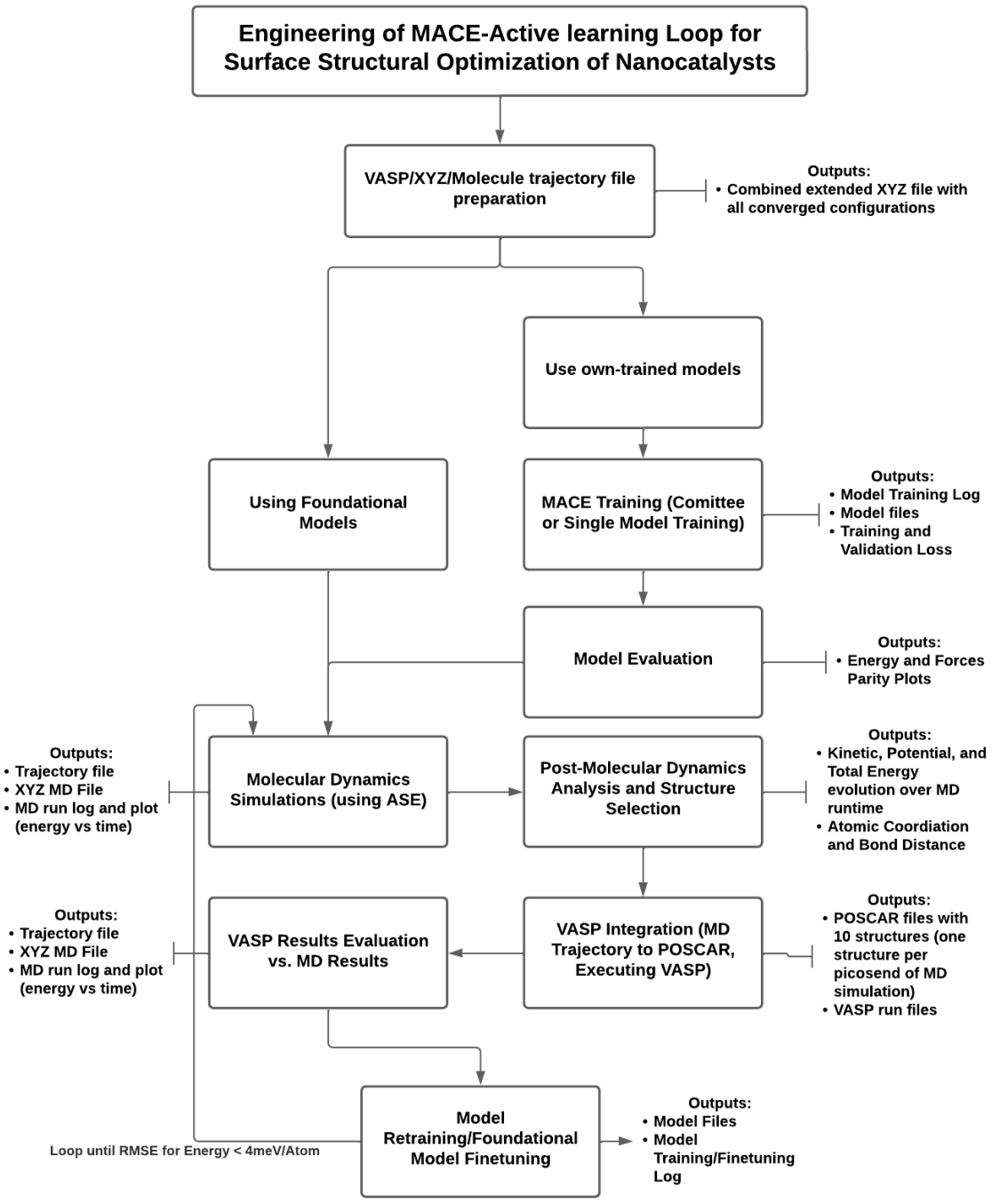


Figure 3.5: MACE Active Learning Loop Engineering

This framework begins by reading VASP output folders and then transforms them into a combined XYZ file for its use in training a single or a committee of models. The user can also choose to instead use a foundational model as its base for the first MD simulation. After the MD simulation analysis, 1 structure is selected every two picoseconds of simulation; however, from the second half of the MD trajectory, one structure is taken every picosecond. This change in taking structures is done due to initial results demonstrating that selecting structures in this range proves to better fit for the model for increased performance in reproducing stable configurations throughout the MD simulation.

These structures are then transformed into POSCAR-readable files for VASP, where a z-threshold is defined to fix the bottom layers of the Ga_2O_3 support. The INCAR, KPOINTS, and POTCAR files must be defined by the user, and the correct directory must be inserted in the code calling the run_vasp.py script. After running VASP, the RMSE comparing energies and forces before and after VASP is calculated. A script named vasp_eval_error was designed for this, where the same optimized configurations from the VASP runs are compared with their POSCAR configurations. Here, an energy error threshold of 0.04 eV/atom is defined to determine if the active learning loop should continue or not. In the case that the error is higher than this value, then the model retraining (for self-trained models) or finetuning (for the foundational model) takes place. The vasp_eval_error also outputs a fine_tuning_dataset.xyz file, where the optimized configurations from the VASP run are found. In the case of retraining a model or a committee of models, this dataset is added to the original training dataset, randomized, and the model is trained once again. In the case of the foundational model, this dataset is directly used as training data for the finetuning script.

For the finetuning to take place, it is necessary to run once again the mace_run_train script. However, this time, additional arguments are required to make sure the foundational model is finetuned correctly [68, 87]. These commands are:

- **pt_train_file**: Path to the replay dataset (set to ”mp” to use directly the Materials Project data)
- **filter_type_pt**: Filtering strategy for the configurations of the pt training file.
- **subselect_pt**: Method for sub-selection.
- **weight_pt**: Weight for the pretraining head loss of the model.
- **num_samples**: Number of samples to use from the replay dataset (set to 6 to exclusively use configurations with only Ni-Ga-O).
- **multiheads_finetuning**: Command to enable multihead finetuning.
- **force_mh_ft_lr**: By default, multihead finetuning uses a lower learning rate (0.0001) and enables EMA. Set to True to override this behavior.

An important argument set here is the multi-head replay finetuning. This is a recently introduced technique in MACE which allows for simultaneously finetuning a foundational model on both a user-defined target dataset and a ”replay” dataset from the foundational model, as shown by the scheme in Figure 3.6. This proves to prevent the catastrophic forgetting and it maintains the model’s generalization capabilities [87]. Initial testing without the use of multihead replay finetuning did show forgetting of the previous datasets, and made the active learning loop more or less useless after the first active learning loop. Appendix D shows an example of this, where the Ni cluster showed the same configuration and similar MD trajectories after finetuning the model without the use of multihead replay. It also made the Ni atoms cluster one on top of each other, which is far from the expected configuration seen after running an MD of this configuration with DFT.

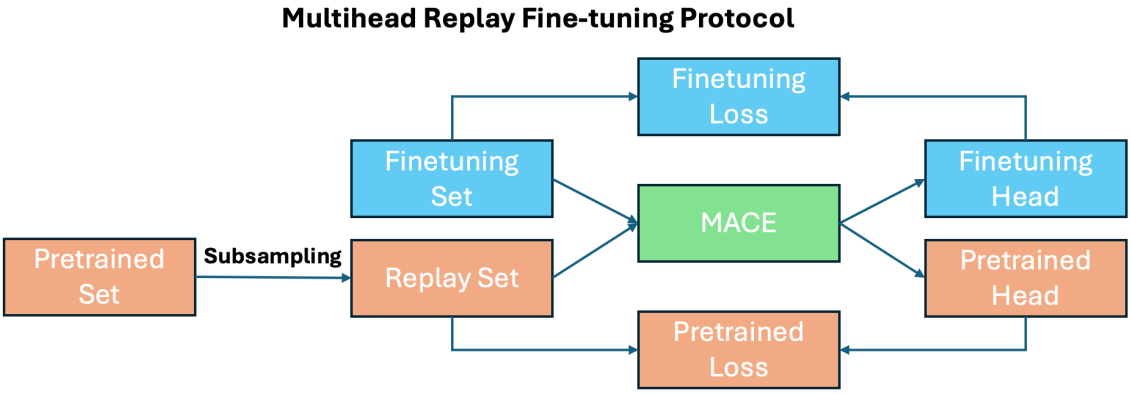


Figure 3.6: Multihead Replay Finetuning Protocol in MACE, obtained from MACE documentation [87]

After the model is retrained or finetuned, the loop begins once more with an MD run. This time, the number of steps is increased by 1 picosecond to allow a larger stability analysis. Afterwards, the loop continues as normal, the model runs VASP again, and determines if the error in energy is low enough to stop the active learning loop. The retraining or fine-tuning datasets update takes into account all of the previous configurations used for the previous model retraining, which prevents any forgetting of the models.

To better visualize the evolution of nickel configurations across multiple active learning loops in MD simulation, two analyses are made: Principal Component Analysis (PCA) and Pair-Distribution Function (PDF) analysis. PCA focuses specifically on Ni-Ni interatomic distances as a structural descriptor, effectively creating a ”structural landscape” where similar configurations cluster together and different structural motifs occupy distinct regions. PCA is used to reduce the high-dimensional distance data into interpretable two-dimensional visualizations that reveal patterns in structural diversity and evolution [100]. When implementing the code, it carefully establishes global y-axis limits across all plots to ensure consistent scaling, enabling direct visual comparison of how structural diversity changes as the machine learning models are iteratively improved. This temporal analysis is particularly valuable for understanding whether active learning successfully explores new regions of structural space or becomes trapped in local minima, and whether the progressive model refinement leads to more confident exploration of previously uncertain structural regions.

Secondly, PDF is applied to assess how accurately the MLIPs models reproduce the local structural environments of nickel atoms compared to the reference VASP DFT calculations, which can directly provide a quantitative measure of model reliability for MD simulations. Here, 10 short MD runs with the same settings but random seeds will be made, and an average of the atomic configurations over the entire trajectory will be taken to calculate the PDF of Ni-Ni and Ni-O bonds. PDFs characterize the statistical distribution of interatomic distances [101]. The designed code systematically computes all pairwise distances between nickel atoms for each timestep, bins these distances into histograms with a resolution of 0.05 Ångströms, and averages across all configurations to produce the final g(r) profile.

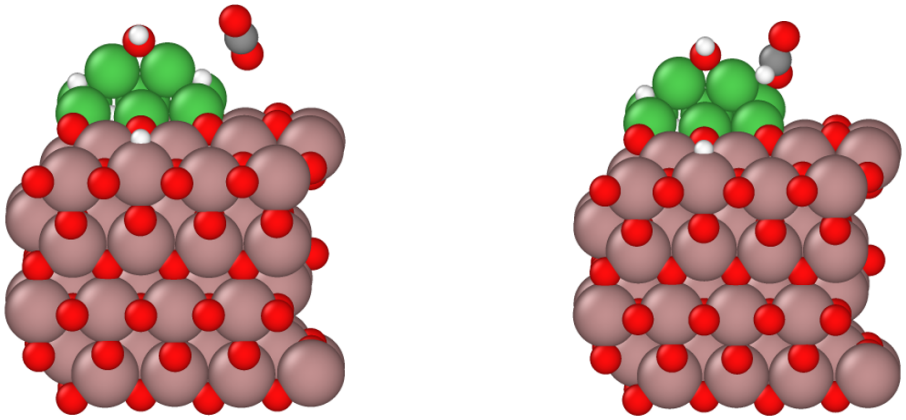
3.4. Energy Barrier for the Formation of Formate Ion

Besides studying the thermal stability of the catalyst and its performance in MD using MLIPs, it is also relevant to process the outputted models from an Active Learning Loop (ALL) in a catalytic model. In this project, focus is placed on doing Nudge Elastic Band (or NEB) simulations to calculate the minimum energy pathway for the transition between two relevant reaction steps.

NEB is directly accessible within the atomic simulation environment (ASE) for its use in Python. Multiple optimizers are available, for example, FIRE, MDmin, BFGS, and others. However, in this case, FIRE was used as the optimizer due to its quick and efficient structure optimization algorithm [102].

To test this, an optimized initial and final configuration was received from the ISE group (see fig 3.7). These structures have already been relaxed and optimized; therefore, they are taken as the initial and final steps of the NEB simulation. An active learning loop using NEB will be done to optimize the NEB performance over time. For this, a new Python script named NEB_test_all.py was developed, where it takes an initial and final configuration (as an xyz or CONTCAR file) as inputs, as well as the number of steps for the NEB simulation. Afterwards, coordination analysis using the avg_coordination.py script is done, and then in a similar fashion as the previous ALL, a VASP is run with the selected configurations (one every two ps, then one every one ps starting from the middle of the simulation). These relaxed structures are then used to finetune the MP0 model, and then the NEB run is done over again until either 10 loops have run or the energy in error is smaller than 43meV/Atom. During each loop, a graph showing the energy versus reaction coordinate is plotted, which can give an idea of whether the model converged or not. This code is available with VASP usage; however, due to time constraints, the single-point calculations were skipped, and the images are directly used as fine-tuning data. This could prove to be an issue in the NEB calculations.

Figure 3.7: Initial and Final NEB Configurations



Initial NEB Image **Final NEB Image**

4

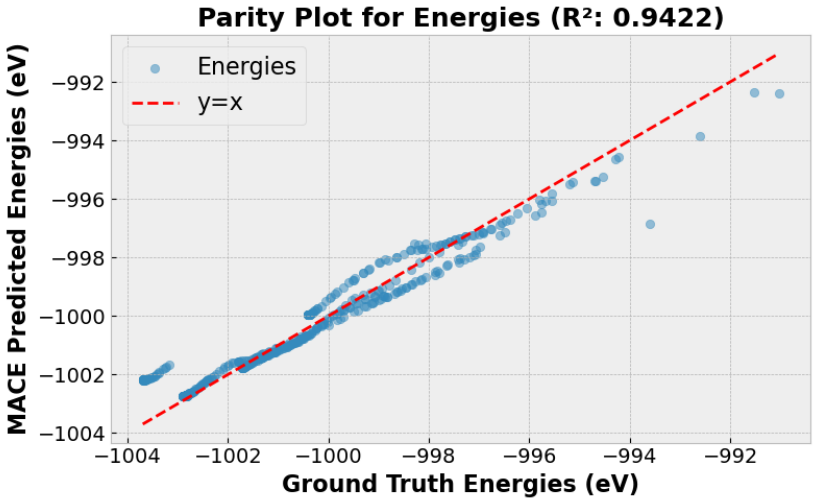
Results and Discussion

Two active learning loops are run for several loops, where one active learning loop uses the MACE-MP0 foundational model as a starting point for the MD simulation, meanwhile the other loop uses self-trained MACE models using the given Ni/Ga_2O_3 system. In this case, a committee of 5 models was trained. To pre-process this data, each model had the same initial dataset, but the data was randomized before the 80/20 training/testing split. This technique is known as cross-validation, which is done to prevent overfitting or bias in the training set, enhancing the overall performance and accuracy of the model [103].

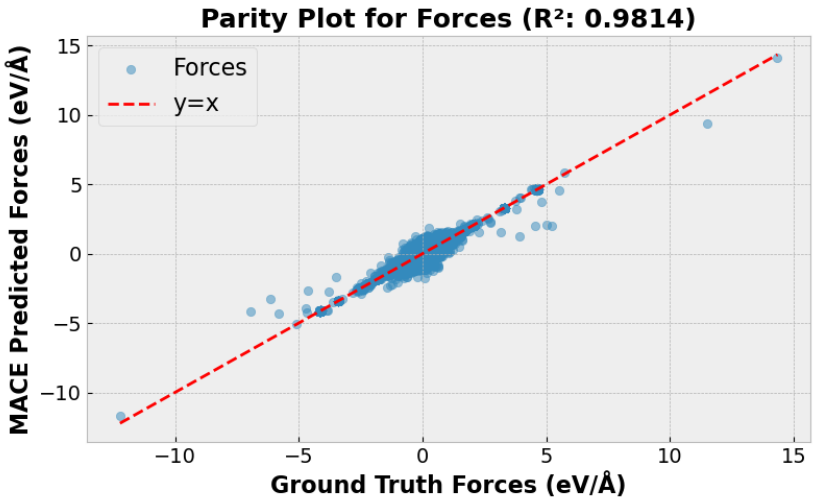
4.1. MACE Training and Validation

For the initial testing and validation with MACE, a model was trained using a single VASP output folder, containing exactly 78 configurations. The results from this training have been shown previously in figure 3.2 and figure 3.3. The parity plot shows a good fit of the predicted energies with the ground-truth DFT energy. However, when evaluating the model with other configurations that were not included in the training set, the model does not perform as well. Parity plots for energies and forces depicting this difference in performance are shown in Figure 4.1. The calculated RMSE for energies and forces are 0.37 eV/atom and 0.0290 ev/Åfor figure 3.2, meanwhile for 3.3 the errors are 5294 and 0.0985 ev/Å. Of course, this is expected since the model is not trained on this data; however, the energy and forces values of the other configurations are around the same values. Therefore, the model might be good enough to do simulations only on the same or similar 78 configurations; however, it might produce errors in energy predictions when applying it in MD with other, non-trained configurations.

Figure 4.1: Parity Plots of Different Structure Evaluated in trained MACE Model



(a) Energy Parity Plot



(b) Force Parity Plot

Exploring the error in energy and forces given via the comparison of the training, validation, and testing datasets is an important metric to determine the performance of the model. Initially, a model was also trained using all of the 464 given configurations, where 80% was used for training (371 configurations) and 20% for testing (the remaining 93). Figure 4.2 shows the result of using 80% of all given configurations. Comparing this image to figure 3.2, a good fit of all datasets is attained, where only one training configuration falls outside of the parity line. Since most configurations fall in the parity line, it is plausible to conclude that the training was successful and that the model has learned successfully the energies and forces per configuration. Another indication of successful machine learning training is shown on the training and validation loss graph. No signs of over- or underfitting are observed, which indicates that the model is robust and performance is expected to be good, since the model generalizes well to the data.

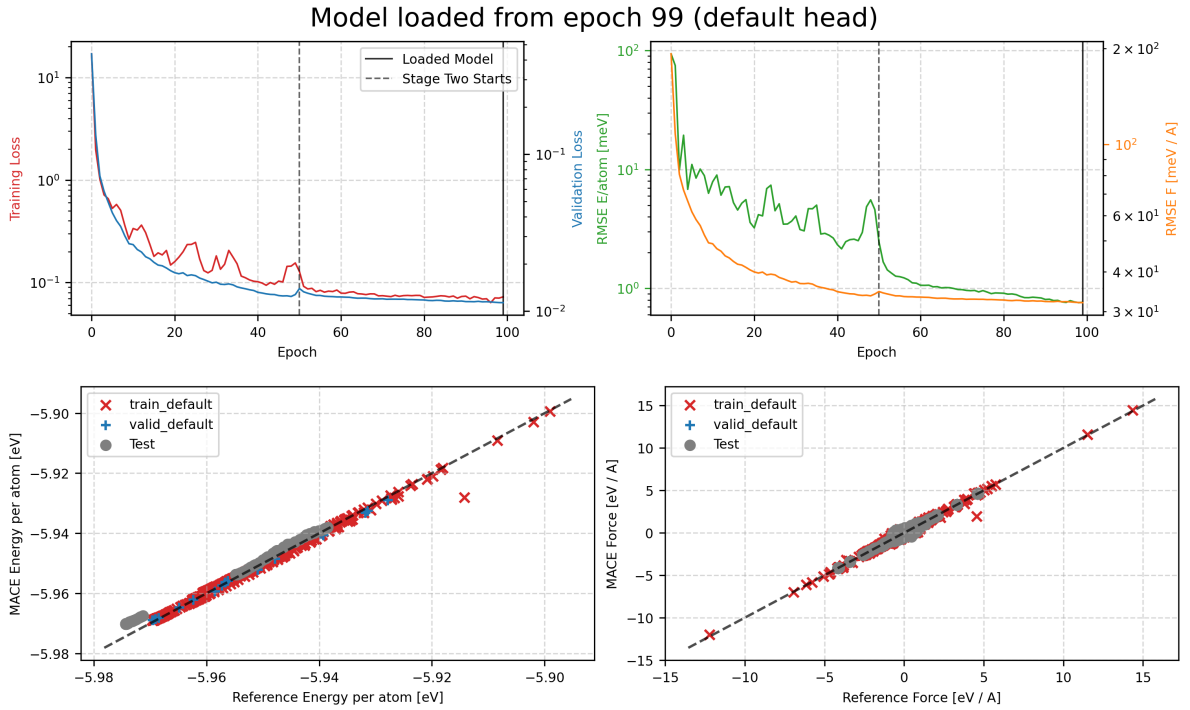


Figure 4.2: MACE Training Results with 371 configurations used for training.

4.1.1. Committee Training

To further explore the evolution of training, validation, and testing error overall, training with a committee of models was also tested. Here, a committee of models with different training sizes was used, where models with 20, 40, 80, 160, 240, and 361 configurations were trained. This was done to test if the error in predicted energy and forces decreases as the number of training points increases. Figure 4.3 and 4.4 show the RMSE for the energies and forces, respectively, for each trained model. These RMSEs are automatically calculated by MACE once the program has finished training. The processing time was calculated from when the MACE training began (epoch 1) until the last epoch of training (epoch 200). All models used the same MACE training settings; the only difference is the number of configurations.

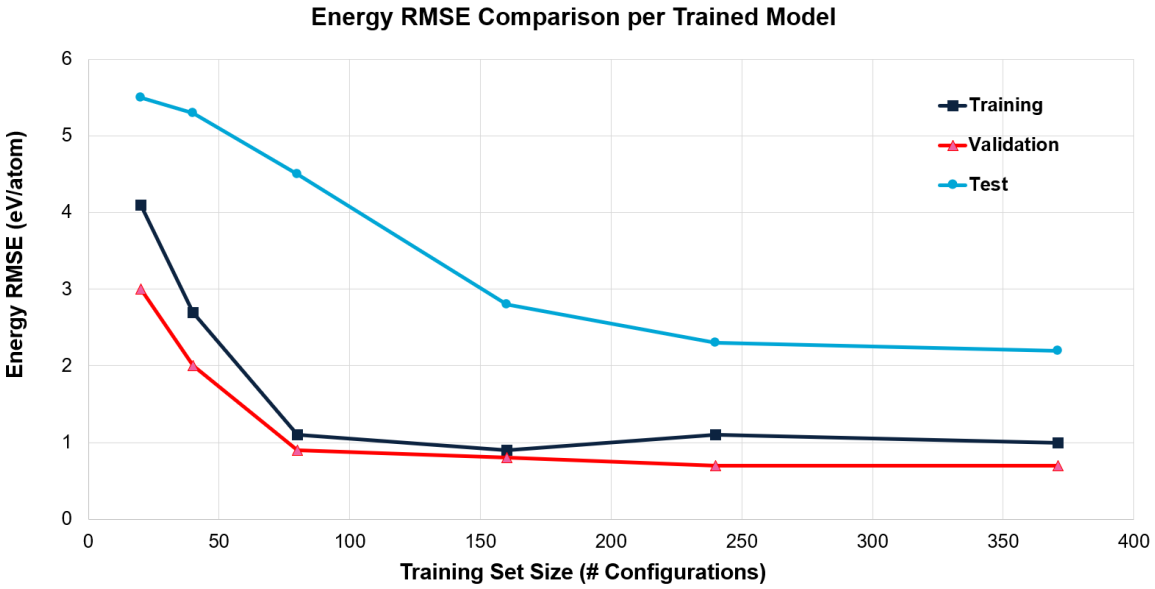


Figure 4.3: MACE Committee Training and Validation Energy Error Evolution

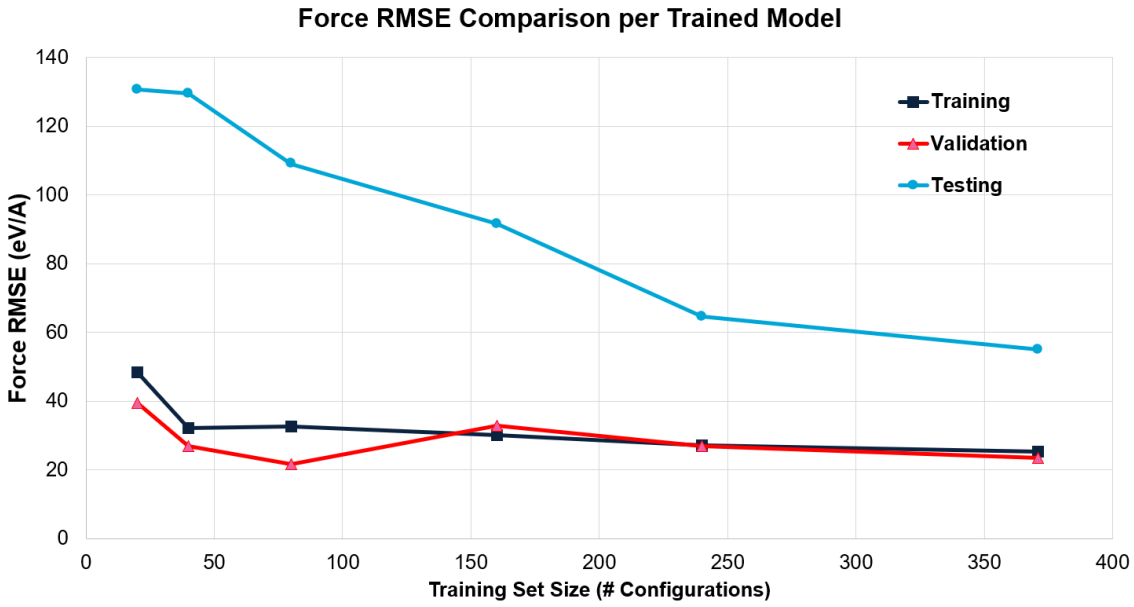


Figure 4.4: MACE Committee Training and Validation Force Error Evolution

As expected, the lowest error in both forces and energy occurs when training on the highest number of structures. However, using 160 and 240 structures could also provide good model performance due to their similarly low errors in energy and forces. Processing time using these models is also shorter due to the lower number of configurations used. Figure 4.5 shows the processing time for the training and evaluation of the models. All trainings and evaluations shown so far have been done locally using one NVIDIA GeForce RTX 4070 graphics card.

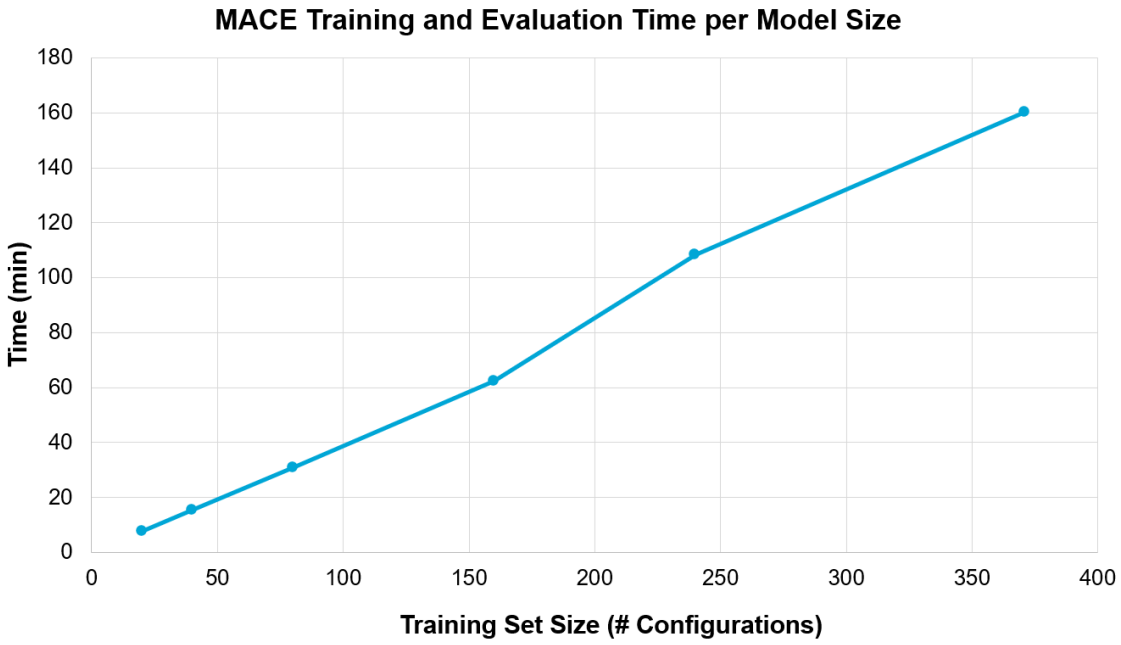


Figure 4.5: MACE Committee Training and Validation Processing time per Model

4.2. Active Learning Loop Results

Firstly, before doing any of the active learning loop simulations, an NVT ensemble with a Langevin dynamics thermostat was set up in VASP. Here, a previously optimized configuration was taken from the same dataset used to train the previous models. This simulation will serve as a basis or "ground truth" to compare the efficacy of the MLIPs-based MD simulations. The INCAR file was set to run for 10ps (the same number of time steps as the initial MD simulations done with the active learning based MD). However, this simulation ran for 885 fs (so 0.885ps) for 24 hours, which is the limit for the educational supercomputer to run. This in itself is a great example of the magnitude of difference in processing requirements, since 10ps using the MP0 model, for example, only takes 10 minutes. These simulations were done using the same number of processors (16 cores, 2 nodes). Figure 4.6 shows the initial and final configuration of this simulation.

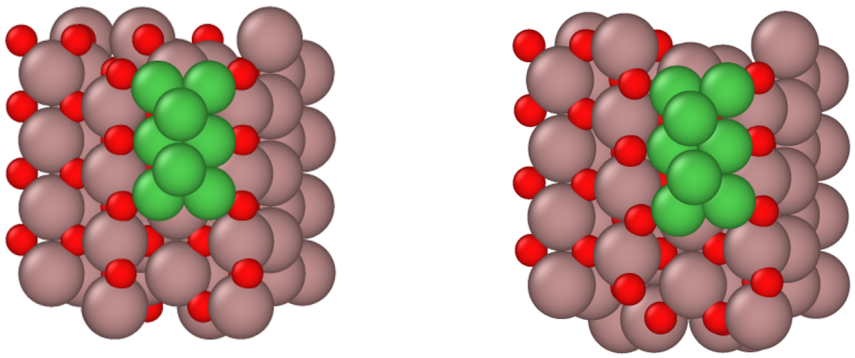


Figure 4.6: Configuration of Ni/Ga_2O_3 (left) before VASP-MD and (right) after VASP-MD

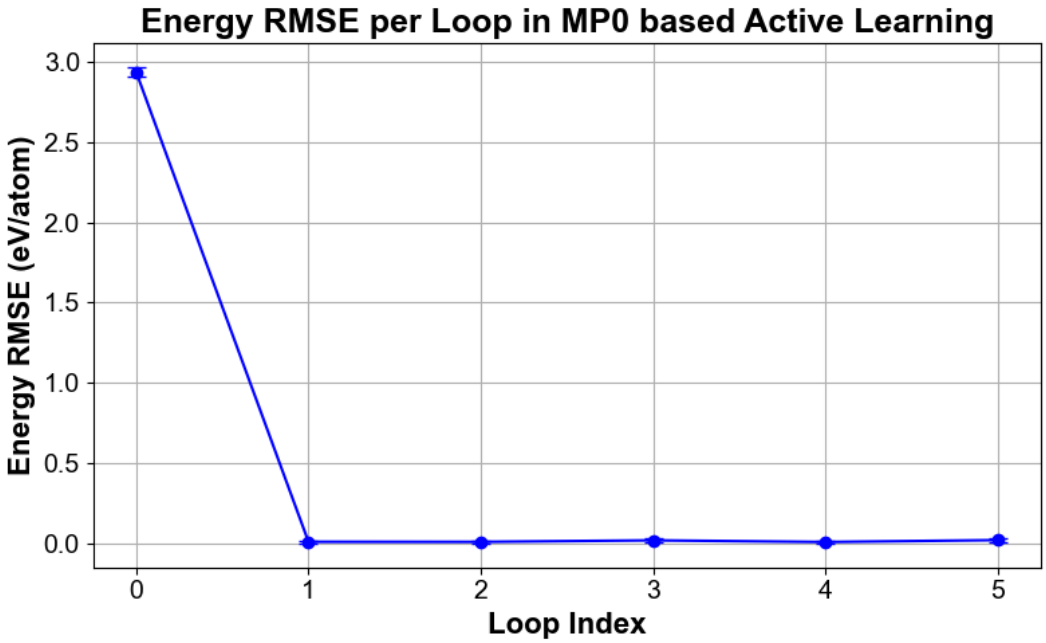
After ending this simulation, the active learning loops using the foundational model MP0 and the committee-based model began. In total, five and three iterations of the loop were done, respectively. For each iteration, all the output files are saved in a folder, which include the MD outputs (such as kinetic, potential, and total energy plotted with the simulation step), coordination analysis plots, trajectory files, and, most importantly, the error in energy between the configurations before and after the single point VASP calculations are printed. It is expected that this error is reduced after each iteration's model finetuning/retraining, since the modified dataset would include configurations of the VASP simulation.

4.2.1. Error Evolution

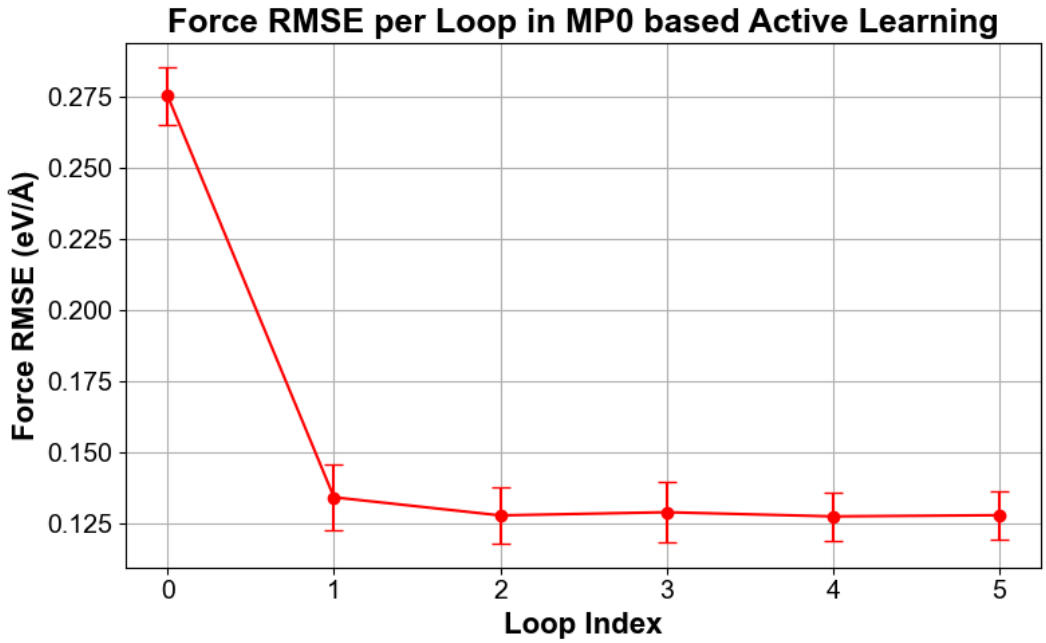
To evaluate the error change in predicted energy and forces, the RMSE of these properties was calculated for the configurations before and after using VASP. For the foundational model-based active learning loop, there was indeed a decrease in energy and force error, though it is reflected more in the energy error evolution. Figure 4.7a shows the energy error evolution per iteration, meanwhile 4.7b shows the force error evolution. The first iteration (before finetuning) reports high RMSE errors of 2.934882 eV/atom and 0.2754 eV/Åfor the energy and force, respectively. After finetuning the model, the error goes down to 0.011099 eV/atom and 0.1297 eV/Å, which proves that finetuning the model for even one iteration reduces the error in energy predictions marginally. The active learning loop would stop right after this since the threshold has been met. However, for testing purposes, three more iterations were done to observe if similar or lower errors were attained.

The following iterations indeed showed similar results in energy error calculations, with the lowest error in energy recorded being of 0.009682 eV/atom, which is already 10meV/atom lower than the established threshold.

Figure 4.7: Error Evolution Before and After Active Learning



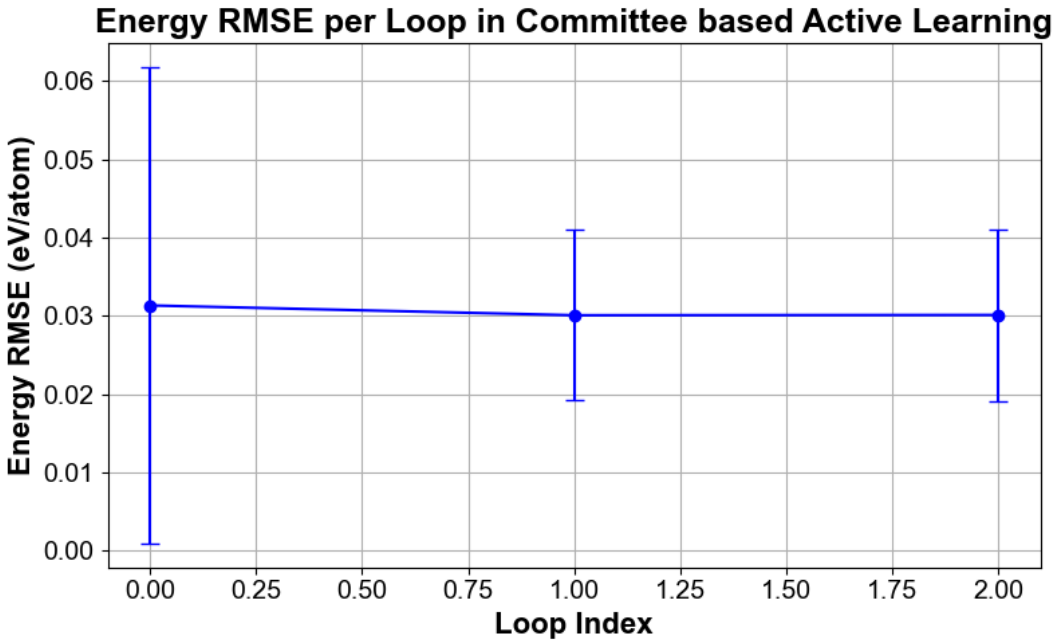
(a) Energy RMSE evolution of MP0 based active learning loop



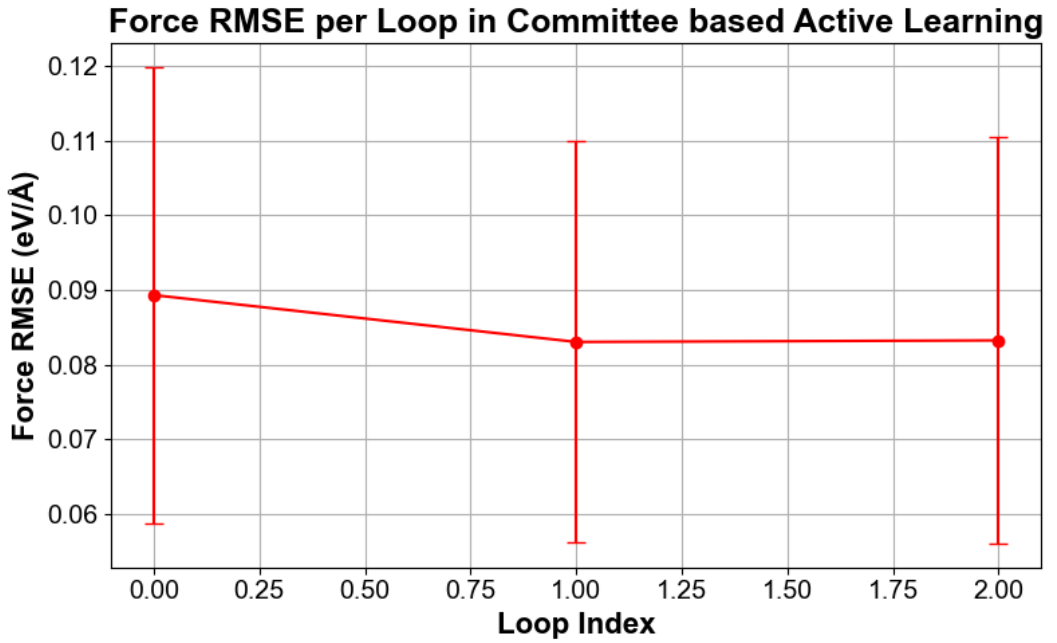
(b) Force RMSE evolution of MP0 based active learning loop

In the case of the committee of models, there is also a decrease in error after the first iteration(see figure 4.8). The error in energy is, however, already below the expected threshold (4eV/atom) before retraining the model. This is expected, since the committee of models is trained exclusively on previously outputted VASP data.

Figure 4.8: Error Evolution Before and After Active Learning with Committee Model



(a) Energy RMSE evolution of Committee-based active learning loop



(b) Force RMSE evolution of Committee-based active learning loop

The difference in error evolution between the types of active learning loops could be because the initial MP0 small foundational model includes information for many configurations of 89 elements. Therefore, when finetuning with only data that includes Ni-Ga-O, the model gives a higher weight to this data. This behavior is not only reflected in a decrease in energy error, but also in the final, stable configuration achieved in the MD simulation. In figure 4.9, the initial, medium, and last configuration of the catalyst are shown, where there is a clear difference in the behavior of Ni atoms on the surface of the support. In

the first scenario, the Ni atoms seem to be clustering one atop of each other. In the other configurations, the Ni atoms create more bonds with the oxygen atoms at the surface of the catalyst, which could be a favored behavior due to a more reproducible, stable, and active catalyst.

The formation of Ni-O bonds can lead to the formation of oxygen vacancies in the Ga_2O_3 support. These then form highly reactive areas that may act as active sites for reactions such as CO_2 hydrogenation to methanol. The CO_2 molecule is then adsorbed on the surface of the catalyst via these active sites. Nickel itself also provides sites for the C-H activation, and this, joined with the previously mentioned formation of oxygen vacancies, creates a synergistic effect allowing for overall better catalytic performances. The higher catalytic performance of nickel supported on various metallic surfaces, such as CeO_2,, is well attributed to weak to medium basic sites, abundant oxygen vacancies, greater dispersion and surface area of nickel, and an appropriate interaction between the metal and the support [104–106]. In the case of the committee-based model, similar stable configurations are achieved throughout all iterations of MD simulations.

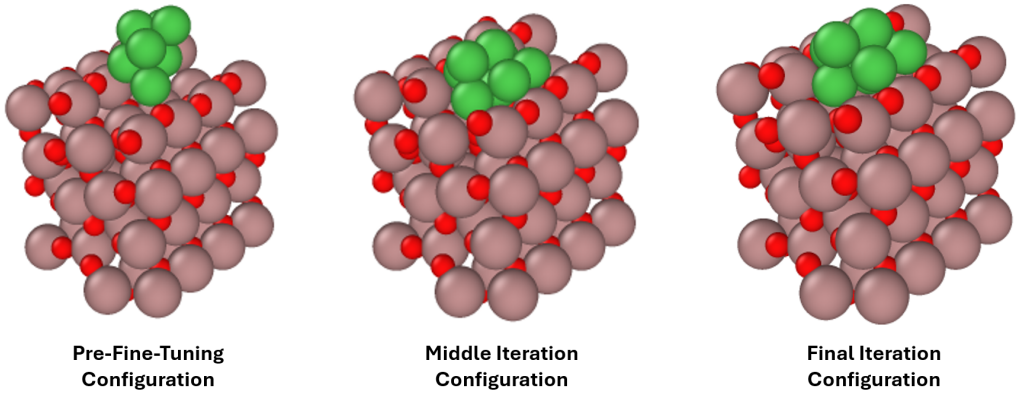


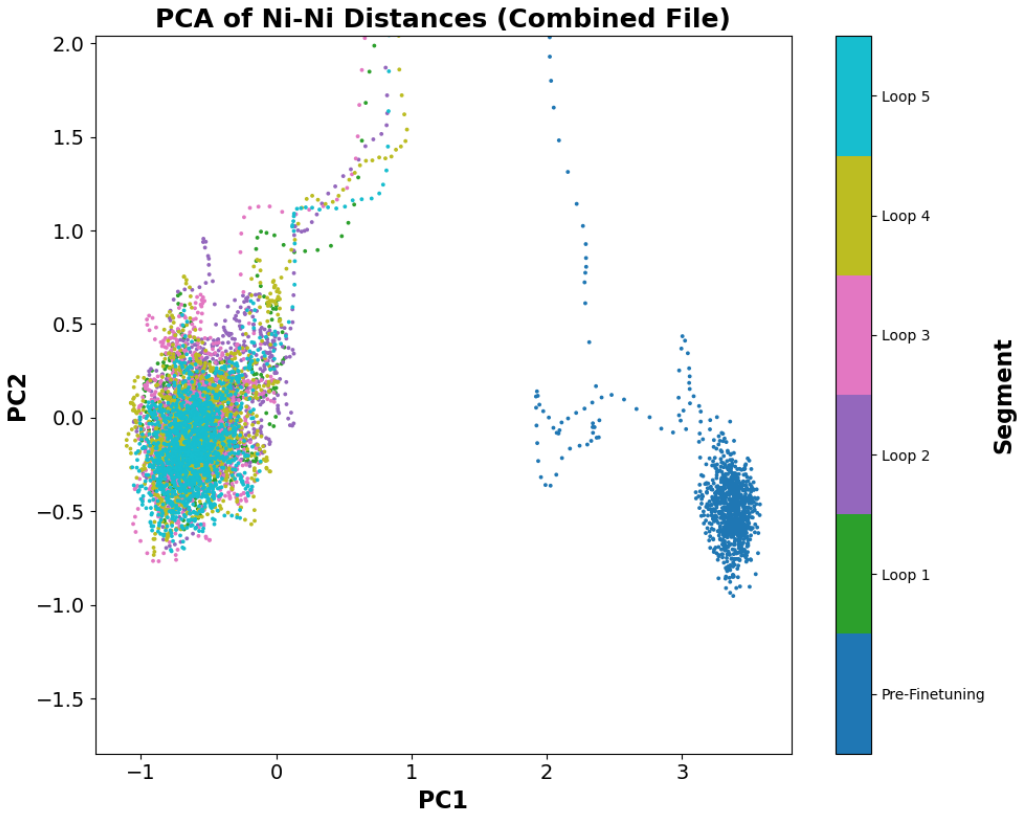
Figure 4.9: Configuration Evolution over MP0 based Active Learning Loop

4.2.2. Principle Component Analysis

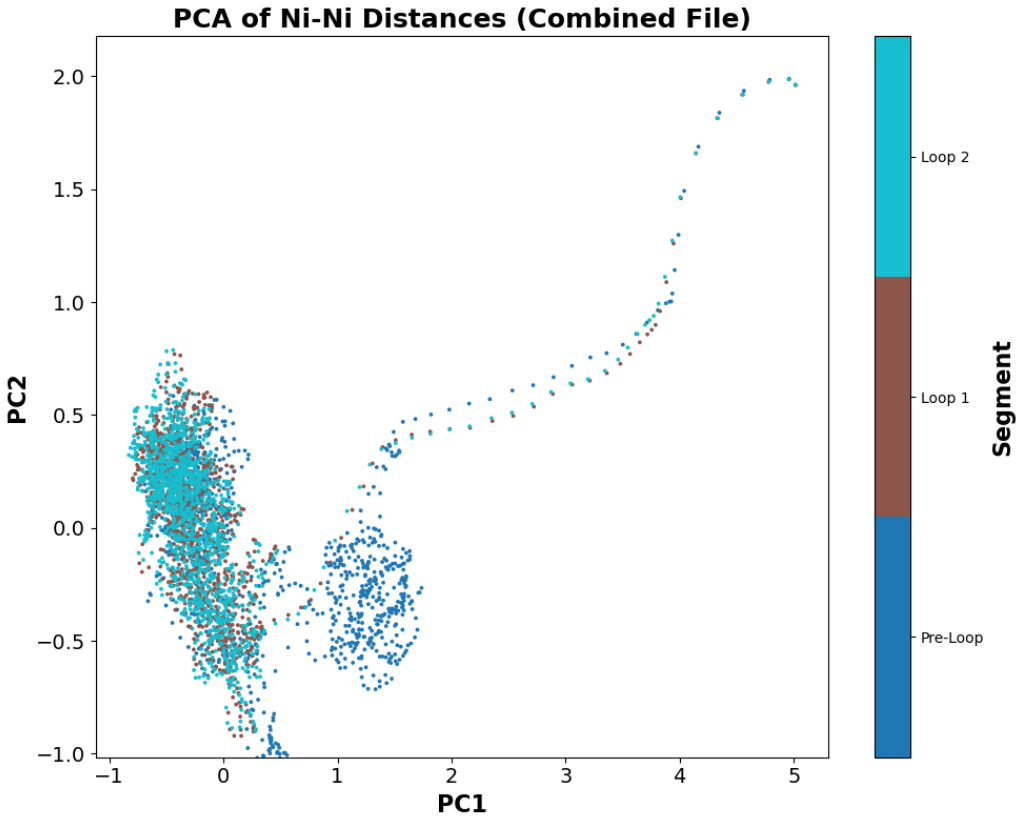
A PCA analysis for both types of ALL was done to identify the variation in data over configurations from all iterations' MD trajectories. Figure 4.10 shows the different PCAs for both types of ALLs. In general, the initial MD configurations begin at high PC2 values and gradually go lower and lower until they all begin to cluster around PC1 and PC2 = 0. The most striking feature of the MP0 PCA is the clear separation between the initial configurations (so pre-fine tuning) with the fine-tuned configurations. This distance between the two clusters suggests that a substantial structural change occurred between the first and second loops of the active learning. This correlates with the previously obtained results on the configuration evolution from the pre-finetuned configuration to the post-finetuning configurations. The fact that all later loops (1-5) occupy the same general area on the plot indicates that after this initial transition, the system had found a stable or preferred conformational basin and continued to sample configurations within that basin.

The initial model generates MD trajectories that are not representative of the catalyst's true conformational space. The large structural differences observed in this PCA suggest that the model was exploring an incorrect part of the potential energy surface. Through fine-tuning, the correct energy landscape was learned by the MACE. This enables the following iterations to accurately explore the true, more stable conformational states of the catalyst. This structural evolution is also observed to a smaller degree on the committee-based active learning loop PCA. Appendix E shows PCA for the entire ALL individually.

Figure 4.10: PCA analysis of all iterations of MP0 and Committee based MD



(a) PCA over all foundational model-based ALL



(b) PCA over all committee models based on ALL

4.2.3. Pair Distribution Function Analyses

To further evaluate the last fine-tuned models, 10 short MD runs of 500 steps are made to study the reproducibility and statistics using said models. The Ni-Ni and Ni-O pair distribution function between the DFT-based MD run and the mean distributions out of the 10 MD simulations for both the foundational-based and committee-based ALLs MDs are compared in figures 4.11-4.14. These graphs show a pretty accurate match between DFT and MLIPs-based predictions, with a 93.75% and 91.25% match in the Ni-Ni pair distribution prediction for the foundational and committee model, respectively. This was calculated using equation 4.1, where the expression calculates the percentage of distance bins where the absolute difference between the VASP DFT reference (g_vasp) and the ensemble-averaged MLIP predictions (g_mlip_mean) falls below a specified tolerance threshold. This provides an intuitive measure of how well the MLIP reproduces the structural characteristics captured by the reference quantum mechanical calculations. The statistical results between both models are shown in Table 4.1.

$$\%\text{match} = \frac{1}{n}\sum_{i=1}^{n}\left(\left|g_{\text{vasp},i} - g_{\text{mlip\_mean},i}\right| < \text{threshold}\right) \times 100 \tag{4.1}$$

Table 4.1: Statistics of PDFs analysis results

Type of Model	Ni-Ni (RMSE)	Ni-O (RMSE)	Match % Ni-Ni (th=0.3)	Match % Ni-O (th=0.1)
MP0	0.1285	0.246	93.75%	68.75%
Committee	0.1401	0.2012	91.25%	70.00%

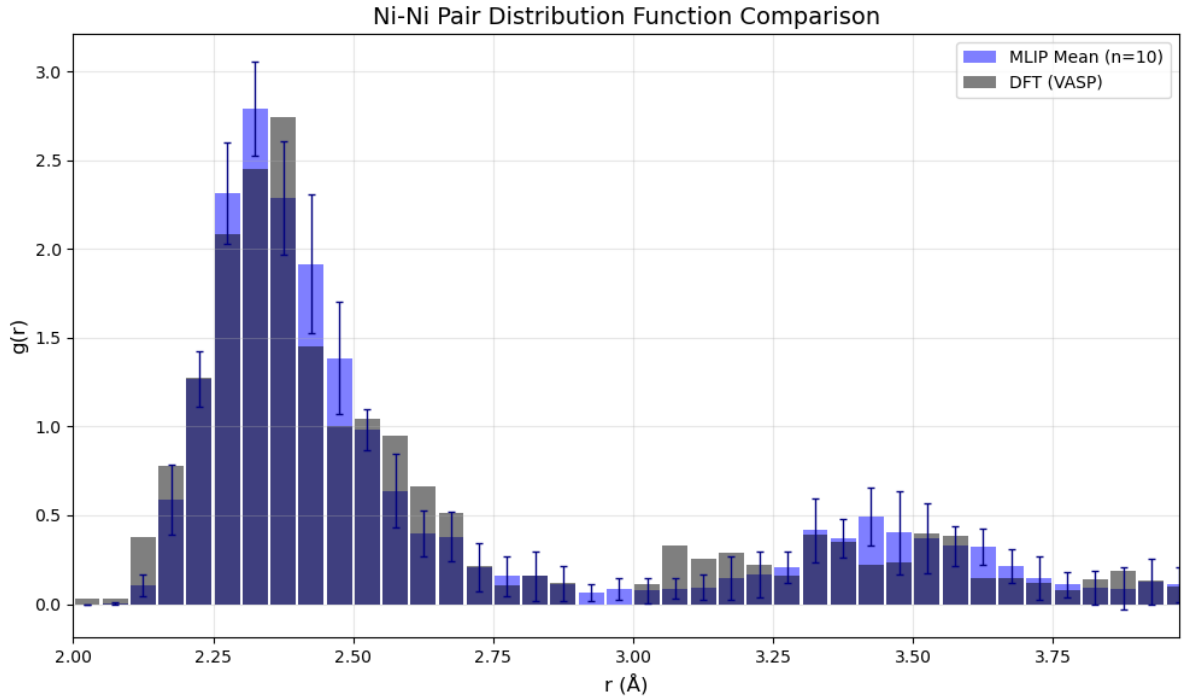


Figure 4.11: Ni-Ni Pair Distribution Function of MP0 and DFT based MD

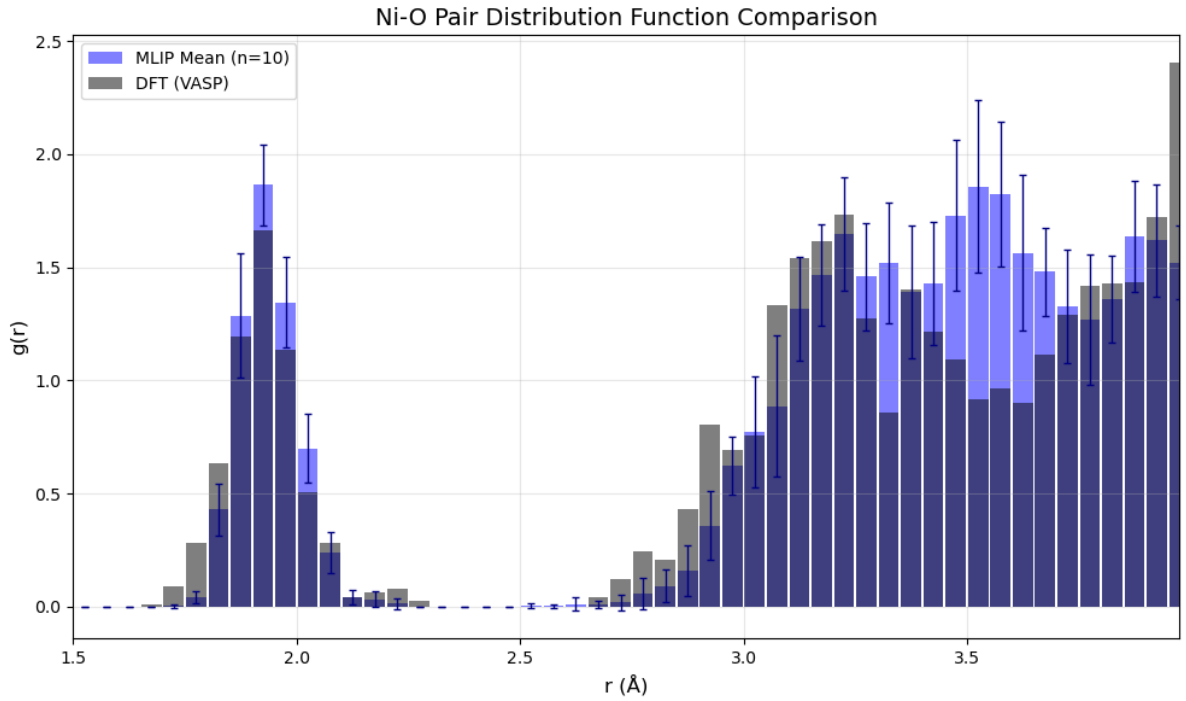


Figure 4.12: Ni-O Pair Distribution Function of MP0 and DFT based MD

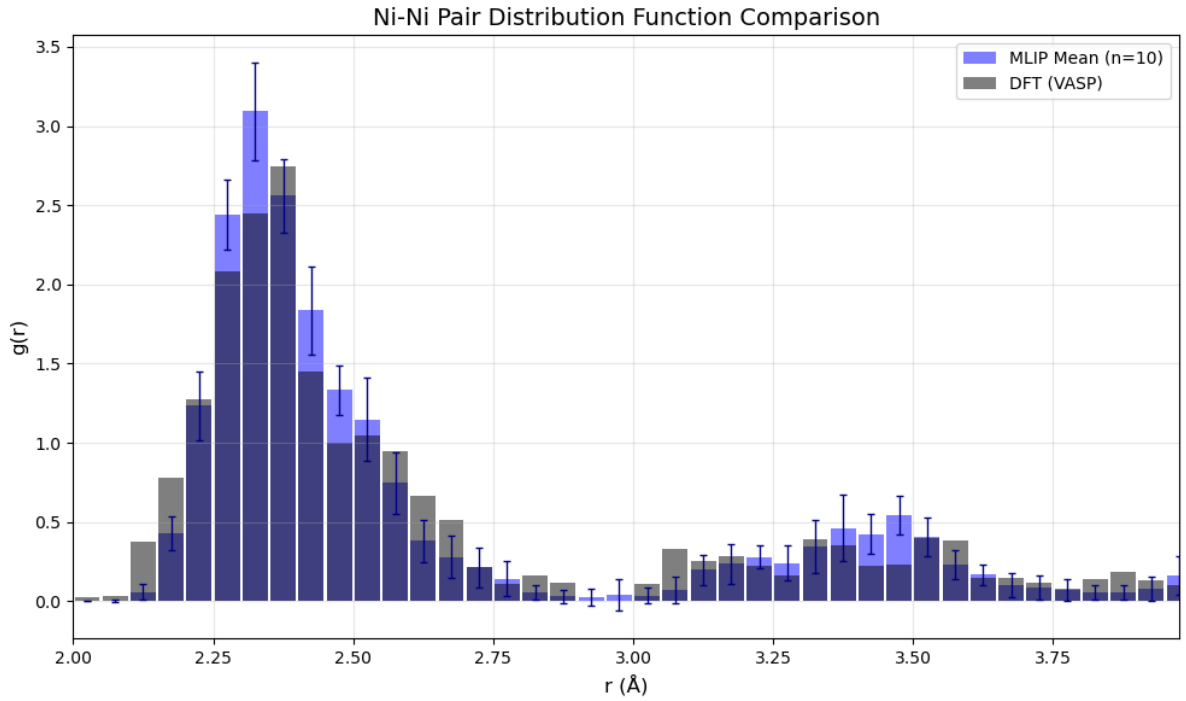


Figure 4.13: Ni-Ni Pair Distribution Function of Committee-based model and DFT-based MD

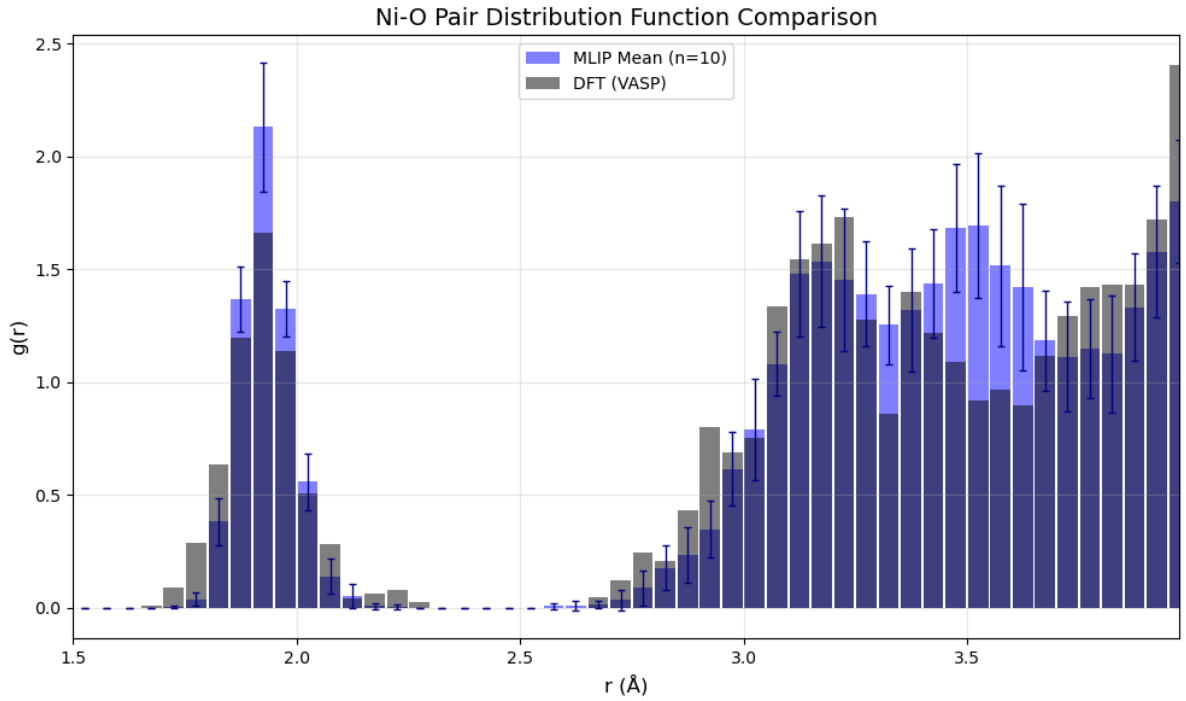


Figure 4.14: Ni-O Pair Distribution Function of Committee-based model and DFT-based MD

Interestingly, MP0-based models show a better match for Ni-Ni interactions compared to committee-based models. In contrast, committee-based models perform better in capturing Ni-O bonding. This improvement is likely a direct result of the committee's ability to identify and sample configurations where the models disagree the most (i.e., configurations with high uncertainty), which often occur at the complex metal-support interface.

However, for Ni-O interactions around 3.5 Å, the predicted density is notably overestimated relative to the DFT ground truth for both loops. One possible explanation is that many GNNs, including MACE, primarily focus on interactions among immediate neighbors. This can lead to descriptors that overlook subtle but important long-range forces. Ni-O interactions beyond the first coordination shell (around 3.0 Å) may not be adequately captured [107].

To address this limitation, recent research has explored hybrid machine learning models that incorporate molecular mechanics methodologies. These approaches treat chemically active regions quantum mechanically while modeling the surrounding environment classically. A notable example is the Field-MACE architecture, which leverages high-order features through multiple expansions. It has demonstrated success in reducing population curve artifacts in nonadiabatic excited-state dynamics. However, its major drawback is computational cost—simulations can take months to complete. Further development is needed to reduce computational time while preserving both short- and long-range interactions in MLIPs [107]. Other distribution functions were calculated (for example, Ni-Ga, Ga-O), as well as the mean coordination number of Ni-Ga. These results for the foundational model-based simulations may be found in Appendix F.

4.2.4. Coordination Distribution

The generalized coordination number (GCN) is another property of interest, which is also calculated in the avg_coordination.py script. Throughout all the iterations for both the MP0 and committee-based active learning loops, an average generalized coordination number in the Ni cluster was found to be

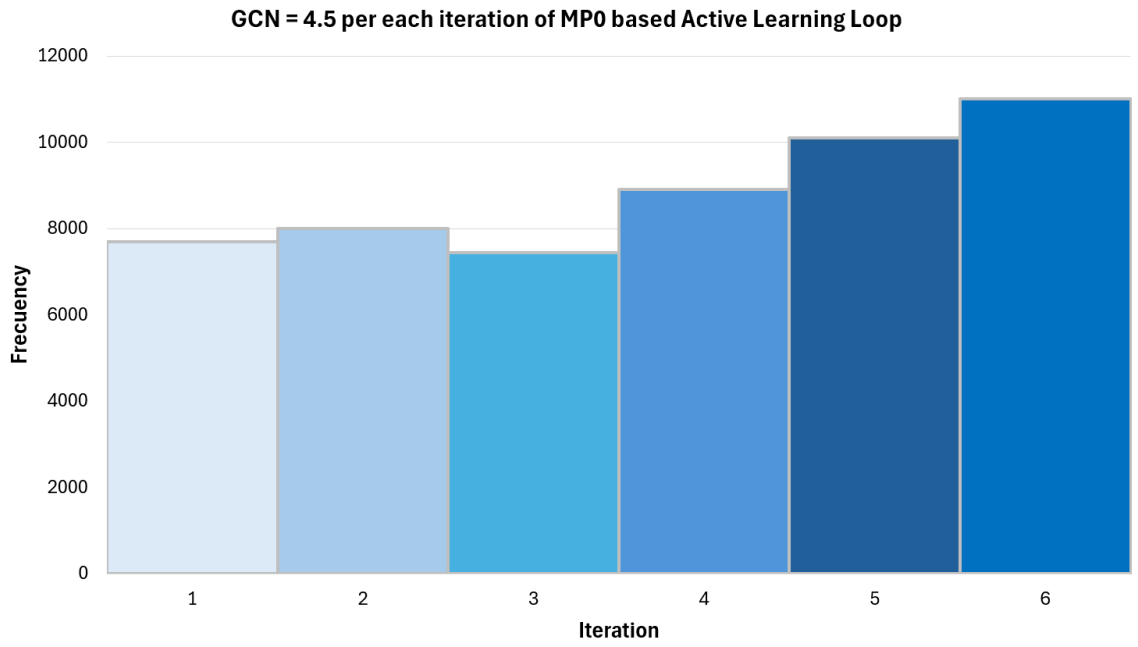
approximately 4 (giving values between 4.09 and 4.15). The generalized coordination number was calculated using equation 4.2. For each nickel atom, the algorithm computes the GCN by summing the normalized coordination numbers of all its neighbors, where each neighbor's contribution is weighted by its coordination number divided by a maximum coordination number (cn_max). This approach captures not only the immediate coordination environment but also the local structural connectivity, providing insights into whether atoms are located in bulk-like environments (high GCN), surface regions (medium GCN), or highly undercoordinated sites such as corners or defects (low GCN). A max coordination number (cn_{max}) of 12 is set due to this being the maximal coordination as found in the bulk for a single atom site in an FCC crystal [98].

A theoretically informed cutoff distance based on the known geometric relationships in FCC crystals, where the nearest-neighbor distances scale as $a/\sqrt{2}$ and next-nearest-neighbor distances scale as the lattice parameter *'a'*. By using a hard-coded lattice parameter of 3.52 Å(typical for nickel) and applying the geometric factor $(\sqrt{2}/2) * 1.45$, the code calculates an adaptive cutoff of approximately 3.6 Åthat is designed to capture both nearest and next-nearest neighbors while excluding more distant coordination shells.

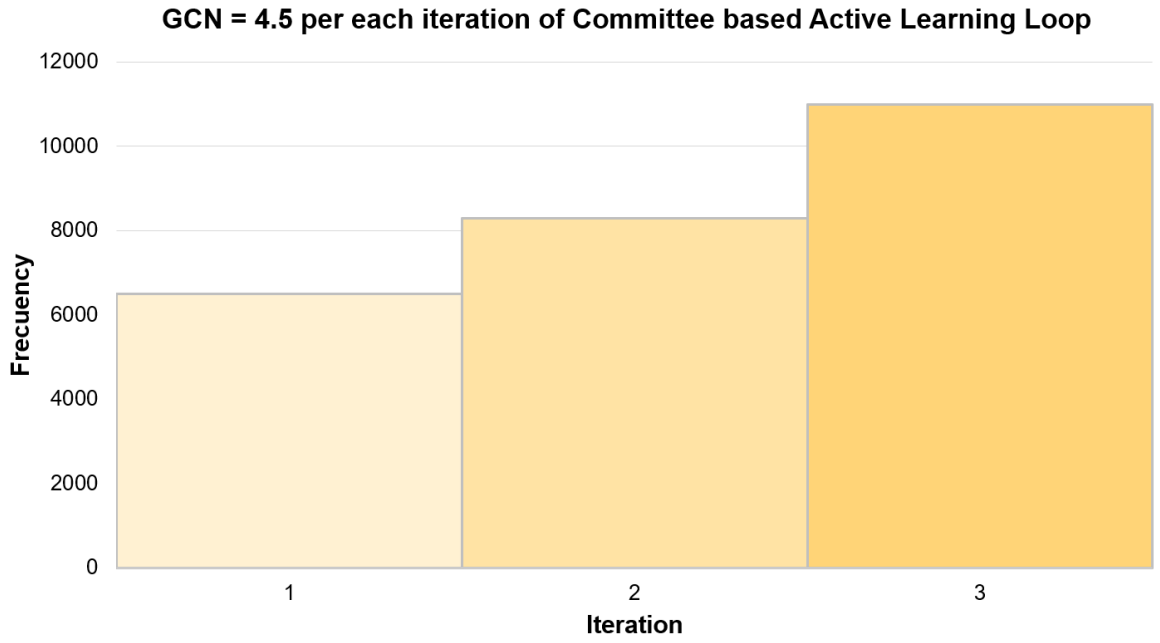
$$\overline{CN}(i) = \sum_{j=1}^{n_i} cn(j)/cn_{max} \tag{4.2}$$

Figure 4.15a and 4.15b show the evolution of the frequency of GCN=4.5 throughout all iterations of the MP0-based model and the committee-based model, respectively. This graph is able to show the evolution of the GCN=4.5 throughout each iteration of active learning, which is the maximum attained GCN seen in every MD trajectory. An increase per iteration is expected and attained (except for iteration 3 of the MP0 iteration), since the timesteps per iteration increased by 1000 (so 1 ps).

Figure 4.15: GCN=4.5 over each iteration in each Active Learning



(a) GCN=4 over each iteration in MP0 based Active Learning



(b) GCN=4.5 over each iteration in committee-based Active Learning

To have a better idea of the model's efficacy in MD, we applied a similar methodology as in the pair distribution function calculations. Here, 10 short simulations were done with the final finetuned and retrained model, respectively. The GCN number is then calculated for both MLIPs and DFT-based MD trajectories, and the mean frequency seen in these trajectories is shown in Figure 4.16. The fact that the GCN for Ni-Ni clusters remains essentially the same in all models suggests that the MLIPs are accurately reproducing the local geometry and bonding topology of the Ni cluster. For this GCN

calculation, an atomic cutoff radius of 3.0 Åwas used. This also supports the previous argumentation about local structural consistency via the usage of MACE and DFT.

The agreement in GCN distributions across both types of MLIPs and DFT models is consistent with the attained match in pair distribution functions for Ni-Ni distances below 3 Å. Since GCN is sensitive to local atomic arrangements, particularly within the first coordination shell, the PDF alignment reinforces the structural fidelity of the MLIPs in capturing short-range order.

This information is especially relevant in catalysis, since it proves that catalytically active sites are preserved regardless of the simulation method. Given the known correlation between low GCN values and enhanced reactivity in Ni-based catalysts, this structural agreement reinforces the reliability of MLIPs in capturing both functional behaviour and stability of the catalyst. Low-coordinated sites have been shown to yield facile C-C coupling barriers and high CO coverage in the CO_2 reduction reaction [98]. This is due to the increase in the number of kinks and step sites, which display enhanced CO binding in copper nanocatalysts [108, 109]. Currently, no research has been found relating to the generalized coordination numbers of different Ni/Ga_2O_3 catalysts. Experimental work coupled with the developed code to generate MLIPs based on DFT single point calculations of Ni/Ga_2O_3 with different coordinative environments, that is, with more or less nickel atoms, would be a great way to relate structure-activity in, say, CO_2 hydrogenation [110].

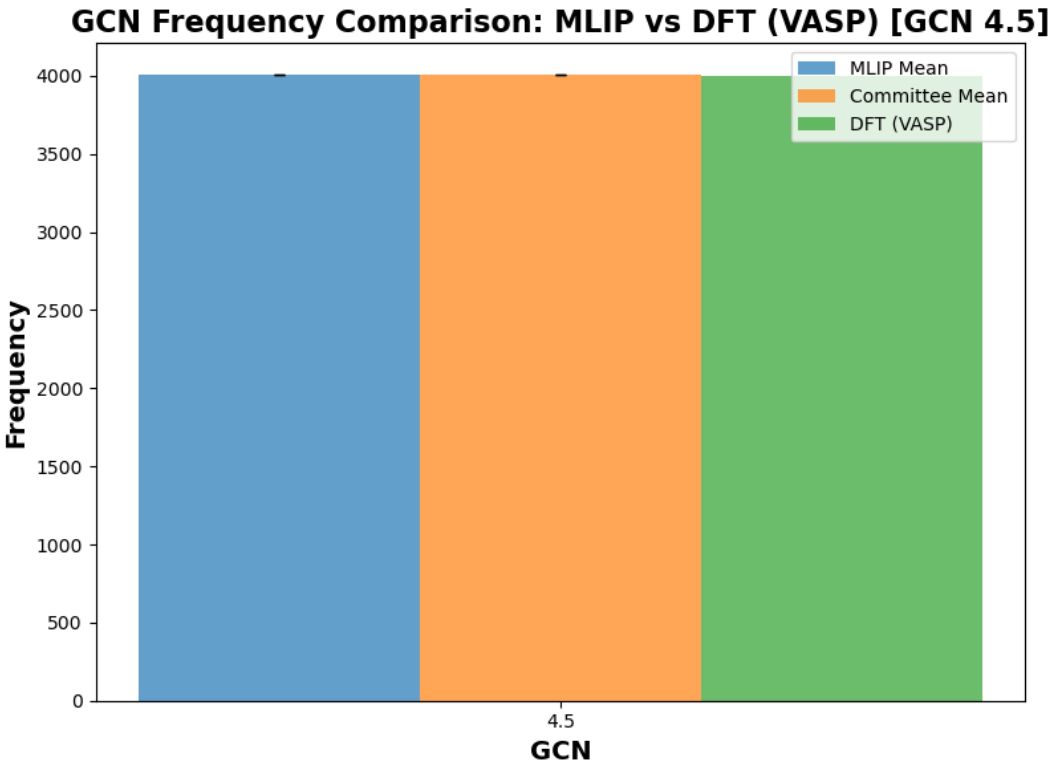


Figure 4.16: GCN = 4.5 Frequency of MLIP and DFT-based configurations

4.3. Catalytic Validation: Energy Barrier of Formate Ion Formation

For the Nudge Elastic Band simulations, the initial and final configurations are shown in fig 3.7. The initial configuration has a CO_2 molecule placed at a certain distance from the Ni cluster. Hydrogen atoms are placed in the Ni cluster, as well as in the Ga_2O_3 support. These hydrogens have been placed there to interact with the CO_2 molecule once it has been adsorbed in the surface of the Ni cluster. The transition step we are interested in studying is the formation of formate. This formation occurs once one of the oxygen atoms from the CO_2 molecule is adsorbed on the Ni surface. Here, an oxygen bridge with the Ni atom is made, and then a hydrogen atom would attach itself to the C atom, resulting in the formation of the formate ion in the Ni cluster surface.

A pre-made DFT-based NEB run of this trajectory has been previously done by the ISE team. Figure 4.17 shows the NEB calculated energies per atomic configuration. Now, the idea is to replicate this behaviour via the usage of MLIPs. For this, the outputted NEB configurations are used as training data to finetune the MACE-MP foundational model.

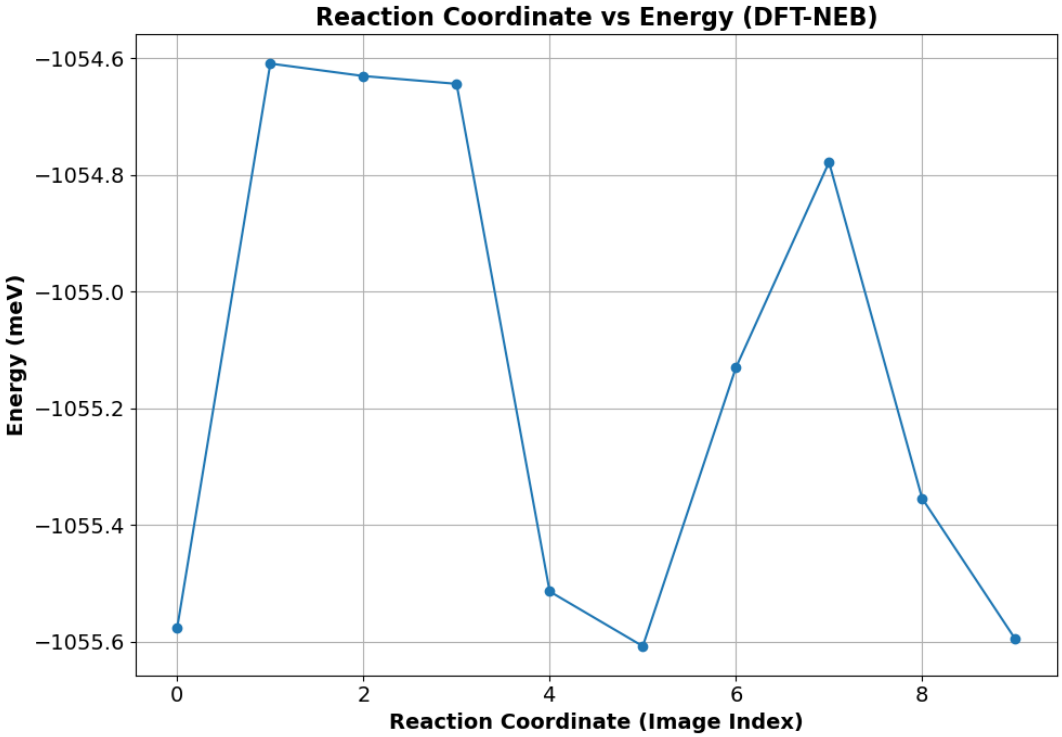


Figure 4.17: Reaction Coordinate of Formate Ion Formation via DFT-NEB

The foundational model is chosen due to the low amount of initial training data at our disposal (only 11 configurations/images are available). These images, however, did not contain information on the forces or energies, and therefore single-point VASP calculations were done with these structures to obtain the relevant properties for proper MACE finetuning. The first MACE-based run was done with the pure foundational model, that is, without being finetuned.

Initially, NEB runs with the foundational model, and subsequent finetuned models were done with 500 steps, with each iteration increasing the timesteps by 100. However, instability in the NEB calculations was observed as the number of timesteps increased. In the first 50-100 timesteps, the MACE-based NEB showed a similar behaviour to the DFT-based NEB run. Therefore, 50 timesteps were chosen as the initial number of steps, and per iteration, it was increased by 10. In total, 10 active learning loops

were done, where in each iteration, an NEB run is done. The output NEB images are then used as fine-tuning data for the following iteration. Following each NEB calculation, the energy versus reaction coordinate (or image index) is made.

To observe the evolution of NEB in a more quantitative perspective, the distance between one of the Ni atoms with the carbon atom of the CO_2 molecule is calculated for every image in the NEB runs. This serves as a plausible calculation to conclude if the MACE-based NEB replicates well (or not) the DFT-based NEB. Figure 4.19 shows the initial results of this simulation.

As seen in the figure, the DFT-based NEB run (blue line with circular markers) shows a smooth, physically intuitive trajectory, with the C-Ni distance gradually decreasing as the molecule adsorbs onto the surface. In contrast, the initial pre-finetuning run (orange line) and subsequent MLIP loop runs (green to blue lines) show significant deviations. The MLIP curves exhibit erratic behavior, particularly at the intermediate images (e.g., indices 2-5) and a sharp, unphysical drop in distance towards the final images (e.g., indices 11-12). This erratic behavior confirms that the MACE-based NEB runs do not accurately replicate the DFT-based trajectory. This is further evidenced in figure 4.18 where the calculated energy per reaction coordinate of all MACE-NEB runs versus the DFT-NEB set. The same deviations are observed in the same reaction coordinates, and also the first and last images have different energy values compared to the DFT and the pre-finetuned NEB runs, which is unexpected. Though this is likely due to the change in trained potential after finetuning.

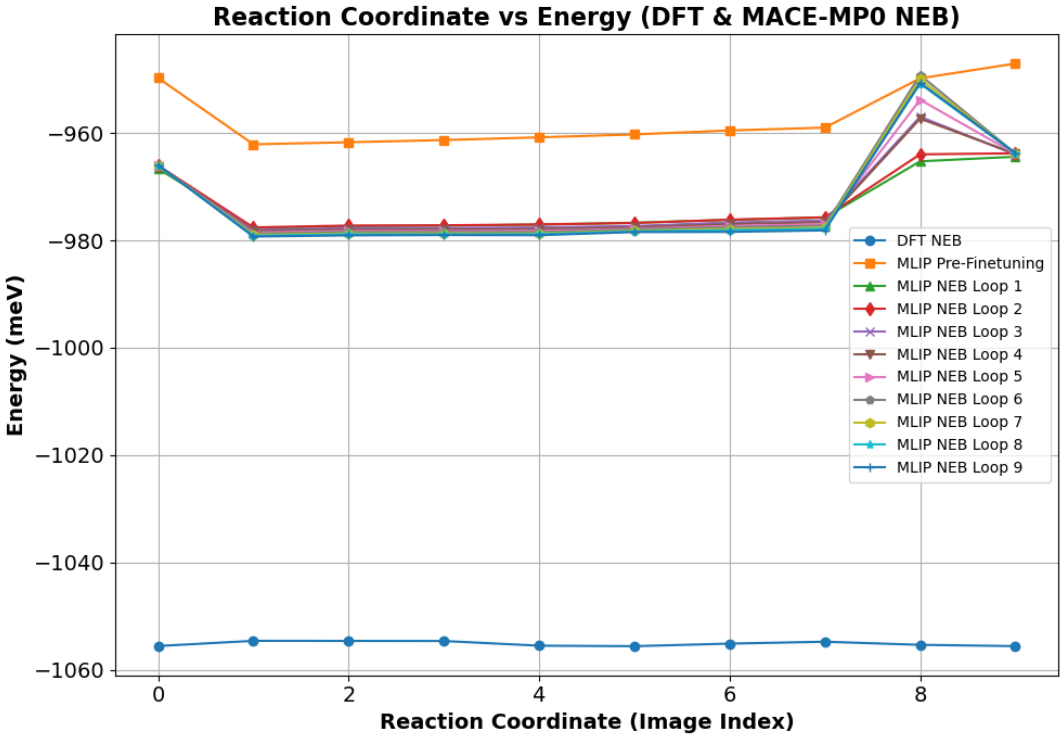


Figure 4.18: Energy per reaction coordinate for each NEB run with MP0

These deviations may arise due to multiple reasons, for example, geometry optimizations (or single point VASP calculations) were not done with the NEB images used for finetuning. This would result in structures having unstable, unoptimized geometries, which would impact the stability of the NEB simulation and subsequent calculations. This is also evidenced by the instabilities seen in the second and second-to-last images. The initial and final images are fixed, and smooth transitions such as those observed in the DFT-based NEB should be observed. However, this is not the case, indicating instabili-

ties in the structure or that the quality of the finetuned MP model is not good enough. Because MACE-MP0's training data doesn't include the specific atomic arrangements and chemical environments found in catalytic systems, it is not well-equipped to provide a realistic description of the reaction pathways and energy barriers in these systems.

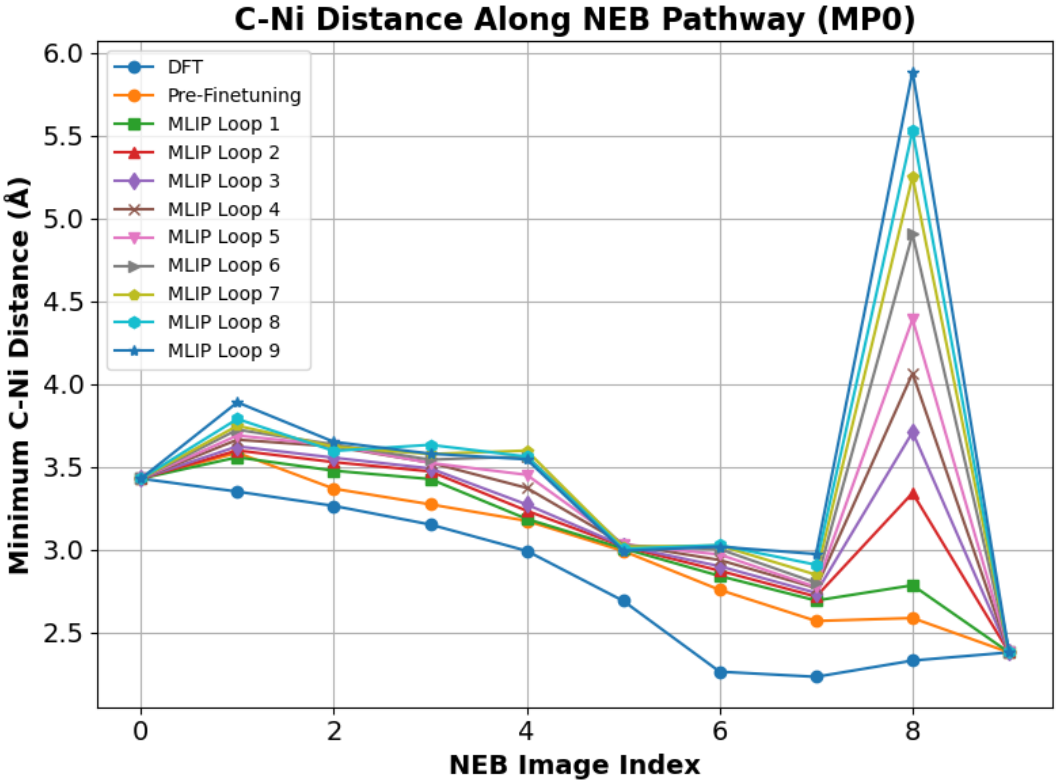


Figure 4.19: C-Ni Distance per reaction coordinate for each NEB run using MP0

Further optimizations should be done before jumping straight into using the foundational model as a calculator in NEB. Correct and plenty of fine-tuning information should be included, as well as doing geometry optimizations in every iteration's training data (NEB images). In this case, this was skipped due to time constraints. A higher number of steps should also be used, since in the DFT-based NEB, 800 steps were used.

As an alternative, I trained a committee of 5 MACE models to use as potentials in NEB. This is done to observe how the performance differs between using the foundational model or a committee of models in an NEB active learning loop. Figure 4.20a shows the reaction coordinate versus recorded energy in the NEB calculation, meanwhile 4.20b shows the C-Ni Distance along the NEB pathway. Comparing these results with figures 4.18 and 4.19 respectively, there is a clear difference in performance altogether.

Figure 4.20a shows the importance of finetuning for energy calculations using MACE. Using a model that was only trained in the initial dataset (so the DFT-NEB) results in wrong energy calculations. Here, energies around -950meV are obtained, meanwhile the DFT results are around -1030 to -1050. This is most likely due to the limited amount of data used for training, resulting in high errors in energy calculations. Of course, this was revised, and indeed, the error in energy decreases with each iteration of the active learning loop.

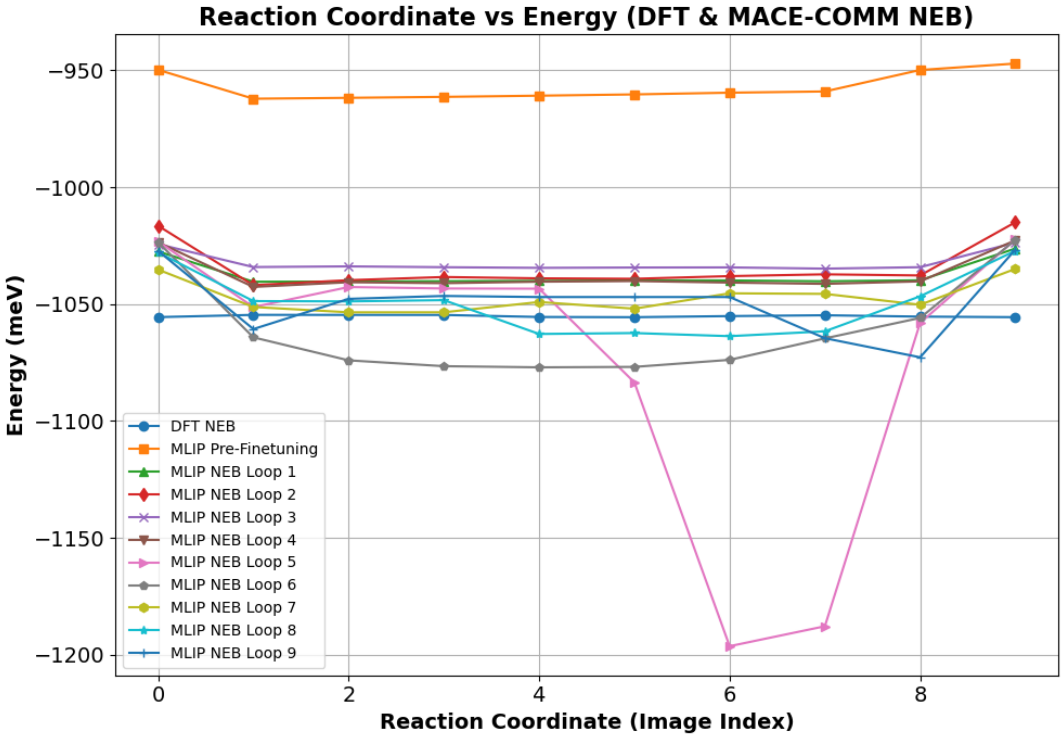
C-Ni curves, while not perfect, generally follow the trend of the DFT curve more closely than the foundational model curves did. The slight deviation is still observed in the first image and second-to-

last image indices; however, the deviation's behavior is different compared to figure 4.19.

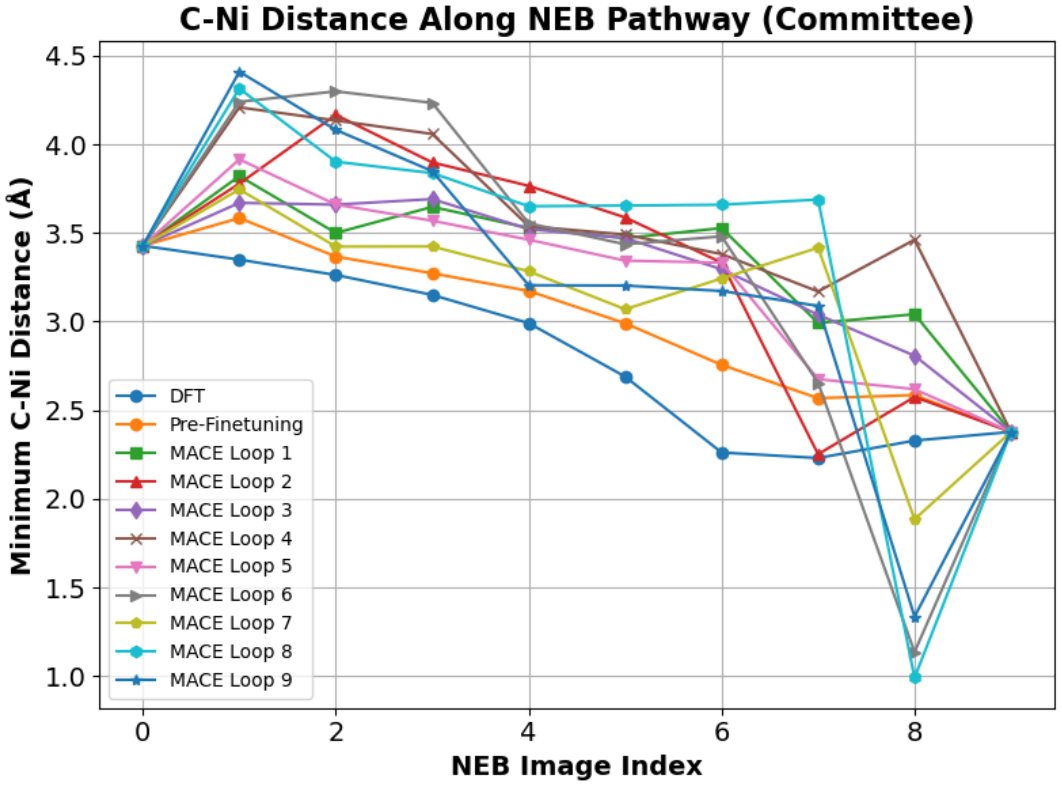
The results of this study indicate that the foundational MACE-MP0 model performs less effectively in NEB simulations compared to the committee-based models. This disparity stems from the foundational model's generalized training on a vast database that does not include the specific atomic arrangements and chemical environments critical for catalytic systems such as the used Ni/Ga_2O_3. Moreover, it lacks the nuanced understanding required to accurately describe the complex reaction pathways and energy barriers, as evidenced by the erratic behavior in both the energy and C-Ni distance plots.

The committee of models, being a self-trained ensemble on catalyst-specific data, provides a better potential for these calculations. Its superior performance underscores the importance of finetuning on domain-specific data. However, even the committee-based model showed some instabilities, suggesting a need for further optimization. One final highlight for the MACE-based NEB calculations is its speed. Only one minute of simulation was required to run 50 steps. Previous simulations, which used 500 steps, took approximately 10 minutes. This was done using. Therefore, 800 steps should take approximately 16 minutes, which is an advantage over the DFT-based run, which took approximately 48 hours, using 192 cores per node, with a total of 232 GB utilized.

Figure 4.20: NEB with a committee of MACE Models



(a) Energy per reaction coordinate for each NEB run with MACE-Committee



(b) C-Ni Distance per reaction coordinate for each NEB run using MACE-Committee

5

Limitations and Future Research

This study provides a foundation for applying MLIPs to a Ni/Ga_2O_3 catalyst system. However, there are several limitations encountered during the realization of this project, which in turn define areas of opportunity for future research studies in these areas. These limitations are the following:

Scope of the trained MLIP models and Element Inclusivity: The MLIPs developed in this work were trained specifically for a pure Ni/Ga_2O_3 system. A significant limitation is the exclusion of other elements, such as hydrogen and carbon. Initially, NEB calculations would not run due to these molecules being missing, which is why it was decided to run additional active learning loops where these elements are indeed included. In the study by Schaaf et al, however, it is shown that six different active learning loops were implemented to sample new training configurations for every loop. These configurations could include MD samples, geometry optimization samples, and NEB samples of the base catalyst material and intermediate states of the material in the CO_2 hydrogenation reaction [51]. This way, the model can understand samples taken from every point of the simulation trajectory and therefore provide accurate results in every step of the simulation.

Training with Stresses: The models on the first shown active learning loops were trained on forces and energies. While this is sufficient for many applications, including the training data with stresses, it would be beneficial for studying phase transformations, high-pressure systems, or performing NPT ensemble MD simulations. This would improve the model's accuracy in predicting structural changes under varying conditions.

Complexity of Reaction Network Modeling: Modeling the entire catalytic reaction network, including all possible reaction intermediates and transition states, is a significant challenge, and worthy of its research article. A comprehensive comparison of the efficacy and accuracy of MLIP-driven methods versus DFT for the entire CO_2 hydrogenation reaction in Ni/Ga_2O_3 would serve as the next step of this investigation, since the base coding for one reaction step is already achieved. This would involve systematically mapping all possible intermediates and transition states. The performance of the MLIPs could then be rigorously benchmarked against traditional DFT calculations, assessing both computational efficiency and accuracy. This could form the basis of a PhD project or a future master's thesis.

Exclusions in Training Data and Model Stability: The foundational MLIP models were trained without including hydrogen due to observed instabilities during initial tests (explosions and unstable configurations). It was therefore decided to only focus on the base Ni/Ga_2O_3. However, it is recommended that this training be tried again, this time with the use of the replay multihead finetuning, since this was missing from the originally trained models with hydrogen. Also, the tested temperature was wrongly set to 1200K, instead of the standard temperatures for these reactions at 300K-600K, so it would be of

interest to try these trainings once more. Due to time constraints, this was not possible in this research project.

Choice of Exchange-Correlation Functional: This study was conducted using data generated at the DFT-PBE level of theory. While PBE is a widely used and well-validated functional for solid-state systems, its accuracy may not be sufficient for all reaction steps, particularly those involving weak interactions or a need for higher precision. Exploring other levels of theory (e.g., meta-GGA, hybrid-GGA, hybrid-meta-GGA functionals) would provide a more robust assessment of the method's performance and is a potential area for future work. This would provide valuable insights into how the choice of functional impacts the MLIP's predictive capabilities and would allow for the creation of more accurate, "higher-level-of-theory" MLIPs.

System Specificity and Code Reusability: The models and scripts designed for this project were tailored specifically to the Ni/Ga_2O_3 system. While the overarching computational framework can be applied to other systems, specific parameters within scripts (for example, coordination analysis and finetuning) are hardcoded for Ni, Ga, and O. To apply this methodology to a different catalyst, these scripts would require significant modification, limiting their immediate "plug-and-play" applicability. A future project could focus on creating a more universal and modular code base that can be easily adapted to different catalytic materials. This would involve generalizing the scripts for coordination analysis and finetuning to automatically handle different elements, thereby enabling the rapid application of this methodology to new systems.

Limitations of the Foundational Model's Dataset: The small MP0a foundational model was used as the basis for finetuning. This model, along with the replay dataset, only included six specific Ni/Ga_2O_3, which was still good enough for the model to favour Ni-O bonding during the simulations. Expanding this foundational dataset to include more diverse structures would be a logical step to improve the model's robustness, though this would necessitate modifications to the current data processing scripts.

Choice of Foundational Model Version: The small MP0a foundational model was used as a base model and foundational finetuning model. However, there are other foundational models which may be used, such as the MP0b versions 1, 2, and 3, which, according to the MACE documentation, provide higher stability in MD simulations, using core repulsion, a new repulsion regularization for high pressure, and a few extra high pressure training example [76]. These models are also recommended to be used in finetuning; however, the base MP0a was used in this research to keep consistency throughout the different simulations.

NEB Calculations Limitations and Performance: For the simulations shown in the NEB results (see section 4.3, performances and results close to the DFT-NEB run were obtained only with the committee-based runs. The primary limitation was the small dataset. Future work should focus on significantly increasing the amount of DFT-NEB-based data. Using a higher number of initial configurations (and also more NEB steps, 800, for example) would give the models a more robust understanding of the reaction pathway. Also, improvement in the quality of the training data at each iteration could be made via VASP calculations for all intermediate configurations generated by the active learning loop. This will provide accurate forces and energies, ensuring that the finetuned model is learning from physically realistic data and mitigating the instabilities observed in the current runs.

6

Conclusion

Machine learning is certainly rising as a plausible complement and even substitute for DFT-based simulations on novel catalyst screening. This study is able to show that both self-trained models and the foundational model MACE-MP0 are great starting points to mimic the DFT behavior in molecular dynamics (MD) simulations. However, with the plain foundational model, the Ni/Ga_2O_3 MD simulations do not depict a similar behavior, where the Ni atoms start to cluster one on top of each other. With the committee of models, this isn't observed. However, after proper finetuning, the foundational model dataset provided very accurate and similar results to the DFT-based MD simulations, obtaining up to a 93% accuracy in Ni-Ni bonding pair distribution function.

This thesis study can show that using MACE is a great alternative to obtaining quick and reliable MD calculations, as well as energetic barrier calculations using NEB. For a simulation with the same Ni/Ga_2O_3, an MD simulation using a NVT ensemble with a Langevin thermostat for 10ps, it takes only 15 minutes with the MACE models. Meanwhile, with the DFT-based simulation in VASP, it would take approximately 272 hours.

Certainly, there were some limitations encountered, mainly due to computational limits and time constraints. This study only focused on training and fine-tuning on Ni/Ga_2O_3 data. Other types of catalysts were not tested; however, with the developed active learning loop pipeline, it would certainly be more accessible to test other systems easily. Long-range interactions are also not well captured by MACE, since this type of ML model primarily focuses on the interactions among the immediate neighbors. Additional development or implementation of molecular mechanics methodologies in MACE, as seen in the Field-MACE architecture, would prove to reduce the miscalculation in these ranges.

Additional testing and code development is also required for the NEB active learning loop would be required as well, since the code up to the date of publishing this thesis shows there are some instabilities when using MACE-MP0 finetuned models as potentials in NEB calculations. Using a committee of models showed better performance compared to the MP0-based NEB calculations. However, deviations are still present, and further testing/fine-tuning is recommended to obtain a model with performance similar to DFT-NEB.

Continued research at the intersection of chemistry, materials science, and data science is essential for addressing these challenges. The integration of these fields is not only accelerating nanocatalyst development but also contributing to a safer, sustainable, and more efficient industrial transition.

References

[1] P. O. o. t. E. Union. “Greenhouse gas emissions from fossil fuel fired power generation systems.” In: (Sept. 2017). ISSN: 1018-5593. URL: https://op.europa.eu/en/publication-detail/-/publication/221658dd-9556-4591-86ea-51544346a8f7.

[2] H. Ritchie, P. Rosado, and M. Roser. “Breakdown of carbon dioxide, methane and nitrous oxide emissions by sector”. In: *Our World in Data* (June 2020). URL: https://ourworldindata.org/emissions-by-sector.

[3] M. J. Orella, Y. Román-Leshkov, and F. R. Brushett. “Emerging opportunities for electrochemical processing to enable sustainable chemical manufacturing”. In: *Current Opinion in Chemical Engineering* 20 (June 2018), pp. 159–167. ISSN: 2211-3398. DOI: 10.1016/J.COCHE.2018.05.002.

[4] G. Centi and S. Perathoner. “Catalysis: Role and challenges for a sustainable energy”. In: *Topics in Catalysis* 52.8 (July 2009), pp. 948–961. ISSN: 10225528. DOI: 10.1007/S11244-009-9245-X/FIGURES/10. URL: https://link.springer.com/article/10.1007/s11244-009-9245-x.

[5] U. Habib, F. Ahmad, M. Awais, N. Naz, M. Aslam, M. Urooj, A. Moqeem, H. Tahseen, A. Waqar, M. Sajid, and M. J. Shabbir. “Sustainable Catalysis: Navigating Challenges and Embracing Opportunities for a Greener Future”. In: *Journal of Chemistry and Environment* 2.2 (Oct. 2023), pp. 14–53. ISSN: 2959-0132. DOI: 10.56946/JCE.V2I2.205. URL: https://www.jspae.com/index.php/jce/article/view/205.

[6] A. I. Osman, A. Ayati, P. Krivoshapkin, B. Tanhaei, M. Farghali, P. S. Yap, and A. Abdelhaleem. “Coordination-driven innovations in low-energy catalytic processes: Advancing sustainability in chemical production”. In: *Coordination Chemistry Reviews* 514 (Sept. 2024), p. 215900. ISSN: 0010-8545. DOI: 10.1016/J.CCR.2024.215900.

[7] E. Gazzarrini, K. Rossi, and F. Baletto. “Born to be different: the formation process of Cu nanoparticles tunes the size trend of the activity for CO2 to CH4 conversion”. In: *Nanoscale* 13.11 (Mar. 2021), pp. 5857–5867. ISSN: 2040-3372. DOI: 10.1039/D0NR07889A. URL: https://pubs.rsc.org/en/content/articlehtml/2021/nr/d0nr07889a%20https://pubs.rsc.org/en/content/articlelanding/2021/nr/d0nr07889a.

[8] C. Jiang, J. Jia, and S. Zhai. “Mechanistic understanding of toxicity from nanocatalysts”. In: *International Journal of Molecular Sciences* 15.8 (Aug. 2014), pp. 13967–13992. ISSN: 14220067. DOI: 10.3390/IJMS150813967. URL: https://www.researchgate.net/publication/264798243_Mechanistic_Understanding_of_Toxicity_from_Nanocatalysts.

[9] K. Rossi, G. G. Asara, and F. Baletto. “A genomic characterisation of monometallic nanoparticles”. In: *Physical Chemistry Chemical Physics* 21.9 (Feb. 2019), pp. 4888–4898. ISSN: 1463-9084. DOI: 10.1039/C8CP05720F. URL: https://pubs.rsc.org/en/content/articlehtml/2019/cp/c8cp05720f%20https://pubs.rsc.org/en/content/articlelanding/2019/cp/c8cp05720f.

[10] M. Núñez, J. L. Lansford, and D. G. Vlachos. “Optimization of the facet structure of transition-metal catalysts applied to the oxygen reduction reaction”. In: *Nature Chemistry 2019 11:5* 11.5 (Apr. 2019), pp. 449–456. ISSN: 1755-4349. DOI: 10.1038/s41557-019-0247-4. URL: https://www.nature.com/articles/s41557-019-0247-4.

[11] X. Zong and D. G. Vlachos. “Reconciling experimental catalytic data stemming from structure sensitivity”. In: *Chemical Science* 14.16 (Apr. 2023), pp. 4337–4345. ISSN: 2041-6539. DOI: 10.1039/D2SC06819B. URL: https://pubs.rsc.org/en/content/articlehtml/2023/sc/d2sc06819b%20https://pubs.rsc.org/en/content/articlelanding/2023/sc/d2sc06819b.

[12] A. S. Eko. “Catalyst Design: A Narrative Study on Artificial Intelligence”. In: (). DOI: 10.5281/ZENODO.10874819. URL: https://zenodo.org/records/10874819.

[13] M. Tamtaji, H. Gao, M. D. Hossain, P. R. Galligan, H. Wong, Z. Liu, H. Liu, Y. Cai, W. A. Goddard, and Z. Luo. “Machine learning for design principles for single atom catalysts towards electrochemical reactions”. In: *Journal of Materials Chemistry A* 10.29 (July 2022), pp. 15309–15331. ISSN: 20507496. DOI: 10.1039/D2TA02039D. URL: https://pubs.rsc.org/en/content/articlehtml/2022/ta/d2ta02039d%20https://pubs.rsc.org/en/content/articlelanding/2022/ta/d2ta02039d.

[14] W. Zhang, J. Sun, H. Wang, and X. Cui. “Recent Advances in Hydrogenation of CO2 to CO with Heterogeneous Catalysts Through the RWGS Reaction”. In: *Chemistry - An Asian Journal* 19.4 (Feb. 2024), e202300971. ISSN: 1861471X. DOI: 10.1002/ASIA.202300971;WGROUP:STRING:PUBLICATION. URL: /doi/pdf/10.1002/asia.202300971%20https://onlinelibrary.wiley.com/doi/abs/10.1002/asia.202300971%20https://aces.onlinelibrary.wiley.com/doi/10.1002/asia.202300971.

[15] P. Schwiderowski, H. Ruland, and M. Muhler. “Current developments in CO2 hydrogenation towards methanol: A review related to industrial application”. In: *Current Opinion in Green and Sustainable Chemistry* 38 (Dec. 2022), p. 100688. ISSN: 2452-2236. DOI: 10.1016/J.COGSC.2022.100688. URL: https://www.sciencedirect.com/science/article/pii/S2452223622001006?casa_token=U4srqx7d_7UAAAAA:yvV_bxg5usgZmm4h9rdpB1cZ9lLkGLvZV34cAPCC2HFnj-srPTea8f839Xk_Rwgx_vrw2oHxrg.

[16] E. Roduner. “Understanding catalysis †”. In: *Chem. Soc. Rev* 43 (2014), p. 8226. DOI: 10.1039/c4cs00210e. URL: www.rsc.org/csr.

[17] K. Kakaei, M. D. Esrafili, and A. Ehsani. “Introduction to Catalysis”. In: *Interface Science and Technology* 27 (Jan. 2019), pp. 1–21. ISSN: 1573-4285. DOI: 10.1016/B978-0-12-814523-4.00001-0.

[18] V. V. Ranade and S. S. Joshi. “Catalysis and Catalytic Processes”. In: *Industrial Catalytic Processes for Fine and Specialty Chemicals* (Jan. 2016), pp. 1–14. DOI: 10.1016/B978-0-12-801457-8.00001-X.

[19] I. Takigawa, K. i. Shimizu, K. Tsuda, and S. Takakusagi. “Machine learning predictions of factors affecting the activity of heterogeneous metal catalysts”. In: *Nanoinformatics* (Jan. 2018), pp. 45–64. DOI: 10.1007/978-981-10-7617-6{_}3/FIGURES/10. URL: https://link.springer.com/chapter/10.1007/978-981-10-7617-6_3.

[20] A. Afugu, C. R. Kwawu, E. Menkah, and E. Adei. “Supported single-atom catalysts in carbon dioxide electrochemical activation and reduction”. In: *Green Sustainable Process for Chemical and Environmental Engineering and Science: Carbon Dioxide Capture and Utilization* (Jan. 2023), pp. 547–560. DOI: 10.1016/B978-0-323-99429-3.00010-2.

[21] K. Rigby and J. H. Kim. “Deciphering the issue of single-atom catalyst stability”. In: *Current Opinion in Chemical Engineering* 40 (June 2023), p. 100921. ISSN: 2211-3398. DOI: 10.1016/J.COCHE.2023.100921.

[22] H. Ooka, J. Huang, and K. S. Exner. “The Sabatier Principle in Electrocatalysis: Basics, Limitations, and Extensions”. In: *Frontiers in Energy Research* 9 (May 2021), p. 654460. ISSN: 2296598X. DOI: 10.3389/FENRG.2021.654460/BIBTEX. URL: www.frontiersin.org.

[23] B. Ni and X. Wang. “Face the Edges: Catalytic Active Sites of Nanomaterials”. In: *Advanced Science* 2.7 (July 2015), p. 1500085. ISSN: 21983844. DOI: 10.1002/ADVS.201500085. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC5115441/.

[24] C. Chen, Z. Zhang, G. Li, L. Li, and Z. Lin. “Recent Advances on Nanomaterials for Electrocatalytic CO 2 Conversion”. In: (2021). DOI: 10.1021/acs.energyfuels.1c00448. URL: https://doi.org/10.1021/acs.energyfuels.1c00448.

[25] K. Ojha, S. Saha, P. Dagar, and A. K. Ganguli. “Nanocatalysts for hydrogen evolution reactions”. In: *Phys. Chem. Chem. Phys* 20 (2018), p. 6777. DOI: 10.1039/c7cp06316d.

[26] Y. Zhang, X. Liu, and W. Wang. “Theoretical Calculation Assisted by Machine Learning Accelerate Optimal Electrocatalyst Finding for Hydrogen Evolution Reaction”. In: *ChemElectroChem* 11.13 (July 2024), e202400084. ISSN: 2196-0216. DOI: 10.1002/CELC.202400084. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/celc.202400084%20https://onlinelibrary.wiley.com/doi/abs/10.1002/celc.202400084%20https://chemistry-europe.onlinelibrary.wiley.com/doi/full/10.1002/celc.202400084?utm_source=chatgpt.com.

[27] H. Chen, X. Liang, Y. Liu, X. Ai, T. Asefa, X. Zou, H. Chen, X. Liang, Y. Liu, X. Ai, X. Zou, and T. Asefa. “2002435 (1 of 32) Active Site Engineering in Porous Electrocatalysts”. In: (2020). DOI: 10.1002/adma.202002435. URL: https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.202002435.

[28] R. Portela, S. Perez-Ferreras, A. Serrano-Lotina, and M. A. Bañares. “Engineering operando methodology: Understanding catalysis in time and space”. In: *Frontiers of Chemical Science and Engineering 2018 12:3* 12.3 (Oct. 2018), pp. 509–536. ISSN: 2095-0187. DOI: 10.1007/S11705-018-1740-9. URL: https://link.springer.com/article/10.1007/s11705-018-1740-9.

[29] J. A. Moulijn, F. Kapteijn, and G. Mul. *Heterogeneous catalysis for chemical engineers : CH 3101*. English. Delft: TU Delft, Faculteit Technische Natuurwetenschappen, afdeling DelftChemTech, 2006. Chap. 286 blz. ; .. cm.

[30] V. Butera and V. Butera. “Density functional theory methods applied to homogeneous and heterogeneous catalysis: a short review and a practical user guide”. In: *Physical Chemistry Chemical Physics* 26.10 (Mar. 2024), pp. 7950–7970. ISSN: 14639076. DOI: 10.1039/D4CP00266K. URL: https://pubs.rsc.org/en/content/articlehtml/2024/cp/d4cp00266k%20https://pubs.rsc.org/en/content/articlelanding/2024/cp/d4cp00266k.

[31] N. Argaman and G. Makov. “Density functional theory: An introduction □ Articles You May Be Interested In Real-space pseudopotential method for computing the vibrational Stark effect Enhanced dielectric response of Zr O 2 upon Ti doping and introduction of O vacancies Density functional theory: An introduction”. In: *J. Chem. Phys* 68 (2000), pp. 69–79. DOI: 10.1119/1.19375. URL: https://doi.org/10.1119/1.19375.

[32] Y. K. Lee. “Density Functional Theory (DFT) Calculations and Catalysis”. In: *Catalysts 2021, Vol. 11, Page 454* 11.4 (Apr. 2021), p. 454. ISSN: 2073-4344. DOI: 10.3390/CATAL11040454. URL: https://www.mdpi.com/2073-4344/11/4/454/htm%20https://www.mdpi.com/2073-4344/11/4/454.

[33] J. K. Nørskov, F. Abild-Pedersen, F. Studt, and T. Bligaard. “Density functional theory in surface chemistry and catalysis”. In: *Proceedings of the National Academy of Sciences of the United States of America* 108.3 (Jan. 2011), pp. 937–943. ISSN: 00278424. DOI: 10.1073/PNAS.1006652108/ASSET/5E0F9BD1-E0A3-40E9-9BBF-DC911461E379/ASSETS/GRAPHIC/PNAS.1006652108FIG09.JPEG. URL: www.pnas.org/cgi/doi/10.1073/pnas.1006652108.

[34] J. Greeley. “Theoretical Heterogeneous Catalysis: Scaling Relationships and Computational Catalyst Design”. In: *Annual Review of Chemical and Biomolecular Engineering* 7.Volume 7, 2016 (June 2016), pp. 605–635. ISSN: 19475438. DOI: 10.1146/ANNUREV-CHEMBIOENG-080615-034413/CITE/REFWORKS. URL: https://www-annualreviews-org.tudelft.idm.oclc.org/content/journals/10.1146/annurev-chembioeng-080615-034413.

[35] T. Bligaard, J. K. Nørskov, S. Dahl, J. Matthiesen, C. H. Christensen, and J. Sehested. “The Brønsted–Evans–Polanyi relation and the volcano curve in heterogeneous catalysis”. In: *Journal of Catalysis* 224.1 (May 2004), pp. 206–217. ISSN: 0021-9517. DOI: 10.1016/J.JCAT.2004.02.034.

[36] J. Cheng, P. Hu, P. Ellis, S. French, G. Kelly, and C. M. Lok. “Bronsted-Evans-Polanyi relation of multistep reactions and volcano curve in heterogeneous catalysis”. In: *Journal of Physical Chemistry C* 112.5 (Feb. 2008), pp. 1308–1311. ISSN: 19327447. DOI: 10.1021/JP711191J. URL: https://pubs.acs.org/sharingguidelines.

[37] S. J. Pitman, A. K. Evans, R. T. Ireland, F. Lempriere, and L. K. McKemmish. “Benchmarking Basis Sets for Density Functional Theory Thermochemistry Calculations: Why Unpolarized Basis Sets and the Polarized 6-311G Family Should Be Avoided”. In: *Journal of Physical Chemistry A* 127.48 (Dec. 2023), pp. 10295–10306. ISSN: 15205215. DOI: 10.1021/ACS.JPCA.3C05573/SUPPL{_}FILE/JP3C05573{_}SI{_}001.XLSX. URL: /doi/pdf/10.1021/acs.jpca.3c05573?ref=article_openPDF.

[38] K. Schwarz, E. Nusterer, P. Margl, and P. E. Blöchl. “Ab initio molecular dynamics calculations to study catalysis”. In: *International Journal of Quantum Chemistry* 61.3 (Jan. 1997), pp. 369–380. ISSN: 1097-461X. DOI: 10.1002/(SICI)1097-461X(1997)61:3<369::AID-QUA2>3.0.CO;2-U. URL: /doi/pdf/10.1002/%28SICI%291097-461X%281997%2961%3A3%3C369%3A%3AAID-QUA2%3E3.0.CO%3B2-U%20https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-461X%281997%2961%3A3%3C369%3A%3AAID-QUA2%3E3.0.CO%3B2-U%20https://onlinelibrary.wiley.com/doi/10.1002/(SICI)1097-461X(1997)61:3%3C369::AID-QUA2%3E3.0.CO;2-U.

[39] Q. Y. Fan, F. Q. Gong, Y. P. Liu, H. X. Zhu, and J. Cheng. “Modeling Dynamic Catalysis at ab Initio Accuracy: The Need for Free-Energy Calculation”. In: *ACS Catalysis* 14.21 (Nov. 2024), pp. 16086–16097. ISSN: 21555435. DOI: 10.1021/ACSCATAL.4C05372/ASSET/IMAGES/LARGE/CS4C05372{_}0005.JPEG. URL: /doi/pdf/10.1021/acscatal.4c05372?ref=article_openPDF.

[40] R. Iftimie, P. Minary, and M. E. Tuckerman. “Ab initio molecular dynamics: Concepts, recent developments, and future trends”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.19 (May 2005), pp. 6654–6659. ISSN: 00278424. DOI: 10.1073/PNAS.0500193102/SUPPL{_}FILE/00193FIG5.JPG. URL: /doi/pdf/10.1073/pnas.0500193102?download=true.

[41] *The VASP Manual - VASP Wiki*. URL: https://www.vasp.at/wiki/index.php/The_VASP_Manual#Support.

[42] J. P. Perdew and K. Schmidt. “Jacob’s ladder of density functional approximations for the exchange-correlation energy”. In: *AIP Conference Proceedings* 577.1 (July 2001), pp. 1–20. ISSN: 0094-243X. DOI: 10.1063/1.1390175. URL: /aip/acp/article/577/1/1/573973/Jacob-s-ladder-of-density-functional.

[43] J. R. Gomes, J. L. Fajín, M. N. D. Cordeiro, C. Teixeira, P. Gomes, R. S. Pillai, G. Novell-Leruth, J. Toda, and M. Jorge. *Density functional treatment of interactions and chemical reactions at interfaces*. July 2013. URL: https://pureportal.strath.ac.uk/en/publications/density-functional-treatment-of-interactions-and-chemical-reactio-2.

[44] L. Piela. “The Molecule Subject to Electric or Magnetic Fields”. In: *Ideas of Quantum Chemistry* (Jan. 2020), pp. 253–335. DOI: 10.1016/B978-0-44-464248-6.00012-0. URL: https://www.sciencedirect.com/science/article/pii/B9780444642486000120.

[45] P. Politzer and J. S. Murray. “The Hellmann-Feynman theorem: a perspective”. In: *Journal of Molecular Modeling* 24.9 (Sept. 2018), pp. 1–7. ISSN: 09485023. DOI: 10.1007/S00894-018-3784-7/FIGURES/1. URL: https://link.springer.com/article/10.1007/s00894-018-3784-7.

[46] S. Wacławek. “Do we still need a laboratory to study advanced oxidation processes? A review of the modelling of radical reactions used for water treatment”. In: *Ecological Chemistry and Engineering S* 28.1 (Mar. 2021), pp. 11–28. ISSN: 18986196. DOI: 10.2478/ECES-2021-0002.

[47] G. W. T. M. J. Frisch. *Citation | Gaussian.com*. URL: https://gaussian.com/citation/.

[48] F. Neese. “The ORCA program system”. In: *Wiley Interdisciplinary Reviews: Computational Molecular Science* 2.1 (Jan. 2012), pp. 73–78. ISSN: 17590876. DOI: 10.1002/WCMS.81.

[49] G. Henkelman, B. P. Uberuaga, and H. Jónsson. “Climbing image nudged elastic band method for finding saddle points and minimum energy paths”. In: *Journal of Chemical Physics* 113.22 (Dec. 2000), pp. 9901–9904. ISSN: 00219606. DOI: 10.1063/1.1329672. URL: https://www.researchgate.net/publication/224877263_A_Climbing_Image_Nudged_Elastic_Band_Method_for_Finding_Saddle_Points_and_Minimum_Energy_Paths.

[50] G. Henkelman and H. Jónsson. “Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points”. In: *Journal of Chemical Physics* 113.22 (Dec. 2000), pp. 9978–9985. ISSN: 00219606. DOI: 10.1063/1.1323224. URL: https://www.researchgate.net/publication/224877460_Improved_Tangent_Estimate_in_the_Nudged_Elastic_Band_Method_for_Finding_Minimum_Energy_Paths_and_Saddle_Points.

[51] L. L. Schaaf, E. Fako, S. De, A. Schäfer, and G. Csányi. “Accurate Energy Barriers for Catalytic Reaction Pathways: An Automatic Training Protocol for Machine Learning Force Fields”. In: *npj Computational Materials* 9.1 (2023). DOI: 10.1038/s41524-023-01124-2. URL: https://arxiv.org/pdf/2301.09931.

[52] J. Gong, M. Chu, W. Guan, Y. Liu, Q. Zhong, M. Cao, and Y. Xu. “Regulating the Interfacial Synergy of Ni/Ga2O3for CO2Hydrogenation toward the Reverse Water-Gas Shift Reaction”. In: *Industrial and Engineering Chemistry Research* 60.26 (July 2021), pp. 9448–9455. ISSN: 15205045. DOI: 10.1021/ACS.IECR.0C05495/SUPPL{_}FILE/IE0C05495{_}SI{_}001.PDF. URL: https://pubs.acs.org/doi/abs/10.1021/acs.iecr.0c05495.

[53] X. Liao, R. Lu, L. Xia, Q. Liu, H. Wang, K. Zhao, Z. Wang, and Y. Zhao. “Density Functional Theory for Electrocatalysis”. In: *Energy & Environmental Materials* 5.1 (Jan. 2022), pp. 157–185. ISSN: 2575-0356. DOI: 10.1002/EEM2.12204. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/eem2.12204%20https://onlinelibrary.wiley.com/doi/abs/10.1002/eem2.12204%20https://onlinelibrary.wiley.com/doi/10.1002/eem2.12204.

[54] L.-H. Mou, T. Han, P. E. S. Smith, E. Sharman, J. Jiang, L.-H. Mou, J. Jiang, T. Han, P. E. S. Smith, and E. Sharman. “Machine Learning Descriptors for Data-Driven Catalysis Study”. In: *Advanced Science* 10.22 (Aug. 2023), p. 2301020. ISSN: 2198-3844. DOI: 10.1002/ADVS.202301020. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/advs.202301020%20https://onlinelibrary.wiley.com/doi/abs/10.1002/advs.202301020%20https://advanced.onlinelibrary.wiley.com/doi/10.1002/advs.202301020.

[55] B. R. Goldsmith, J. Esterhuizen, J. X. Liu, C. J. Bartel, and C. Sutton. “Machine learning for heterogeneous catalyst design and discovery”. In: *AIChE Journal* 64.7 (July 2018), pp. 2311–2323. ISSN: 1547-5905. DOI: 10.1002/AIC.16198. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/aic.16198%20https://onlinelibrary.wiley.com/doi/abs/10.1002/aic.16198%20https://aiche.onlinelibrary.wiley.com/doi/10.1002/aic.16198.

[56] A. K. Pimachev and S. Neogi. “First-Principles Prediction of Electronic Transport in Experimental Semiconductor Heterostructures via Physics-Based Machine Learning”. In: *npj Computational Materials* 7.1 (Nov. 2020). DOI: 10.1038/s41524-021-00562-0. URL: http://arxiv.org/abs/2011.08426%20http://dx.doi.org/10.1038/s41524-021-00562-0.

[57] L. Zhang, J. Han, H. Wang, R. Car, and E. Weinan. “Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics”. In: *Physical Review Letters* 120.14 (Apr. 2018), p. 143001. ISSN: 10797114. DOI: 10.1103/PHYSREVLETT.120.143001/FIGURES/5/THUMBNAIL. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.120.143001.

[58] M. Erdem Günay and R. Yıldırım. “Recent advances in knowledge discovery for heterogeneous catalysis using machine learning”. In: *Catalysis Reviews* 63.1 (Jan. 2021), pp. 120–164. ISSN: 15205703. DOI: 10.1080/01614940.2020.1770402. URL: https://www.tandfonline.com/doi/abs/10.1080/01614940.2020.1770402.

[59] T. Talaei Khoei and N. Kaabouch. “Machine Learning: Models, Challenges, and Research Directions”. In: *Future Internet 2023, Vol. 15, Page 332* 15.10 (Oct. 2023), p. 332. ISSN: 1999-5903. DOI: 10.3390/FI15100332. URL: https://www.mdpi.com/1999-5903/15/10/332/htm%20https://www.mdpi.com/1999-5903/15/10/332.

[60] M. Kulichenko, B. Nebgen, N. Lubbers, J. S. Smith, K. Barros, A. E. A. Allen, A. Habib, E. Shinkle, N. Fedik, Y. W. Li, R. A. Messerly, and S. Tretiak. “Data Generation for Machine Learning Interatomic Potentials and Beyond”. In: *Chemical Reviews* 124.24 (2024), pp. 13681–13714. DOI: 10.1021/ACS.CHEMREV.4C00572/ASSET/IMAGES/LARGE/CR4C00572{_}0008.JPEG. URL: /doi/pdf/10.1021/acs.chemrev.4c00572?ref=article_openPDF.

[61] G. Wang, C. Wang, X. Zhang, Z. Li, J. Zhou, and Z. Sun. “Machine learning interatomic potential: Bridge the gap between small-scale models and realistic device-scale simulations”. In: *iScience* 27.5 (May 2024), p. 109673. ISSN: 2589-0042. DOI: 10.1016/J.ISCI.2024.109673. URL: https://www.sciencedirect.com/science/article/pii/S2589004224008952.

[62] G. Wang, C. Wang, X. Zhang, Z. Li, J. Zhou, and Z. Sun. “Machine learning interatomic potential: Bridge the gap between small-scale models and realistic device-scale simulations”. In: *iScience* 27.5 (2024), p. 109673. DOI: 10.1016/J.ISCI.2024.109673. URL: https://www.sciencedirect.com/science/article/pii/S2589004224008952.

[63] R. Jacobs et al. “A practical guide to machine learning interatomic potentials – Status and future”. In: *Current Opinion in Solid State and Materials Science* 35 (2025). DOI: 10.1016/j.cossms.2025.101214. URL: http://arxiv.org/abs/2503.09814%20http://dx.doi.org/10.1016/j.cossms.2025.101214.

[64] M. Pinheiro, F. Ge, N. Ferré, P. O. Dral, and M. Barbatti. “Choosing the right molecular machine learning potential”. In: *Chemical Science* 12.43 (Nov. 2021), pp. 14396–14413. ISSN: 2041-6539. DOI: 10.1039/D1SC03564A. URL: https://pubs.rsc.org/en/content/articlehtml/2021/sc/d1sc03564a%20https://pubs.rsc.org/en/content/articlelanding/2021/sc/d1sc03564a.

[65] O. A. Montesinos López, A. Montesinos López, and J. Crossa. “Fundamentals of Artificial Neural Networks and Deep Learning”. In: *Multivariate Statistical Machine Learning Methods for Genomic Prediction* (2022), pp. 379–425. DOI: 10.1007/978-3-030-89010-0{_}10. URL: https://link.springer.com/chapter/10.1007/978-3-030-89010-0_10.

[66] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky. “E(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials”. In: *Nature Communications 2022 13:1* 13.1 (Dec. 2021), pp. 1–11. DOI: 10.1038/s41467-022-29939-5. URL: http://arxiv.org/abs/2101.03164%20http://dx.doi.org/10.1038/s41467-022-29939-5.

[67] S. Takamoto, S. Izumi, and J. Li. “TeaNet: Universal neural network interatomic potential inspired by iterative electronic relaxations”. In: *Computational Materials Science* 207 (May 2022), p. 111280. ISSN: 0927-0256. DOI: 10.1016/J.COMMATSCI.2022.111280. URL: https://www.sciencedirect.com/science/article/pii/S0927025622000799.

[68] I. Batatia, D. P. Kovács, G. N. Simm, C. Ortner, and G. Csányi. “MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields”. In: *Advances in Neural Information Processing Systems* 35 (June 2022). ISSN: 10495258. URL: https://arxiv.org/pdf/2206.07697.

[69] L. Wang, X. Chen, Y. Du, Y. Zhou, Y. Gao, and W. Cui. “CataLM: empowering catalyst design through large language models”. In: *International Journal of Machine Learning and Cybernetics* (Jan. 2025), pp. 1–11. ISSN: 1868808X. DOI: 10.1007/S13042-024-02473-0/TABLES/6. URL: https://link.springer.com/article/10.1007/s13042-024-02473-0.

[70] S. Takamoto et al. “Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements”. In: *Nature Communications 2022 13:1* 13.1 (May 2022), pp. 1–11. ISSN: 2041-1723. DOI: 10.1038/s41467-022-30687-9. URL: https://www.nature.com/articles/s41467-022-30687-9.

[71] A. Merchant, S. Batzner, S. S. Schoenholz, M. Aykol, G. Cheon, and E. D. Cubuk. “Scaling deep learning for materials discovery”. In: *Nature* 624.7990 (Dec. 2023), pp. 80–85. ISSN: 14764687. DOI: 10.1038/S41586-023-06735-9;TECHMETA=119,141;SUBJMETA=1034,1039,117,301,639,705;KWRD=COMPUTER+SCIENCE,SCALING+LAWS. URL: https://www.nature.com/articles/s41586-023-06735-9.

[72] H. Yang et al. “MatterSim: A Deep Learning Atomistic Model Across Elements, Temperatures and Pressures”. In: (May 2024). URL: https://arxiv.org/pdf/2405.04967.

[73] F. Xie, T. Lu, S. Meng, and M. Liu. “GPTFF: A high-accuracy out-of-the-box universal AI force field for arbitrary inorganic materials”. In: *Science Bulletin* 69.22 (Nov. 2024), pp. 3525–3532. ISSN: 20959281. DOI: 10.1016/j.scib.2024.08.039. URL: http://arxiv.org/abs/2402.19327%20http://dx.doi.org/10.1016/j.scib.2024.08.039.

[74] Y. L. Liao, B. Wood, A. Das, and T. Smidt. “EquiformerV2: Improved Equivariant Transformer for Scaling to Higher-Degree Representations”. In: *12th International Conference on Learning Representations, ICLR 2024* (June 2023). URL: https://arxiv.org/pdf/2306.12059.

[75] M. Neumann, J. Gin, B. Rhodes, S. Bennett, Z. Li, H. Choubisa, A. Hussey, and J. Godwin. “Orb: A Fast, Scalable Neural Network Potential”. In: (Oct. 2024). URL: https://arxiv.org/pdf/2410.22570v1.

[76] I. Batatia et al. “A foundation model for atomistic materials chemistry”. In: (Dec. 2023). URL: https://arxiv.org/pdf/2401.00096.

[77] Y. Park, J. Kim, S. Hwang, and S. Han. “Scalable Parallel Algorithm for Graph Neural Network Interatomic Potentials in Molecular Dynamics Simulations”. In: *Journal of Chemical Theory and Computation* 20.11 (Feb. 2024), pp. 4857–4868. DOI: 10.1021/acs.jctc.4c00190. URL: http://arxiv.org/abs/2402.03789%20http://dx.doi.org/10.1021/acs.jctc.4c00190.

[78] C. Chen and S. P. Ong. “A universal graph deep learning interatomic potential for the periodic table”. In: *Nature Computational Science* 2.11 (Nov. 2022), pp. 718–728. ISSN: 26628457. DOI: 10.1038/S43588-022-00349-3; SUBJMETA=1034,1035,1037,301,639; KWRD=ATOMISTIC+MODELS,COMPUTATIONAL+METHODS,MATERIALS+SCIENCE. URL: https://www.nature.com/articles/s43588-022-00349-3.

[79] B. Deng, P. Zhong, K. J. Jun, J. Riebesell, K. Han, C. J. Bartel, and G. Ceder. “CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling”. In: *Nature Machine Intelligence* 5.9 (Sept. 2023), pp. 1031–1041. ISSN: 25225839. DOI: 10.1038/S42256-023-00716-3; SUBJMETA=1034,1035,1042,2790,301,4077,639,705; KWRD=ATOMISTIC+MODELS,COMPUTATIONAL+SCIENCE,ENERGY+MODELLING. URL: https://www.nature.com/articles/s42256-023-00716-3.

[80] K. Choudhary, B. DeCost, L. Major, K. Butler, J. Thiyagalingam, and F. Tavazza. “Unified Graph Neural Network Force-field for the Periodic Table”. In: *Digital Discovery* 2.2 (Sept. 2022), pp. 346–355. DOI: 10.1039/D2DD00096B. URL: http://arxiv.org/abs/2209.05554%20http://dx.doi.org/10.1039/D2DD00096B.

[81] H. Yu, M. Giantomassi, G. Materzanini, J. Wang, and G.-M. Rignanese. “Systematic assessment of various universal machine-learning interatomic potentials”. In: *Materials Genome Engineering Advances* 2.3 (Mar. 2024). ISSN: 2940-9489. DOI: 10.1002/mgea.58. URL: https://arxiv.org/pdf/2403.05729.

[82] B. Focassio, L. P. M. Freitas, and G. R. Schleder. “Performance Assessment of Universal Machine Learning Interatomic Potentials: Challenges and Directions for Materials’ Surfaces”. In: *ACS Applied Materials and Interfaces* 17.9 (May 2024), pp. 13111–13121. DOI: 10.1021/acsami.4c03815. URL: http://arxiv.org/abs/2403.04217%20http://dx.doi.org/10.1021/acsami.4c03815.

[83] B. Deng, Y. Choi, P. Zhong, J. Riebesell, S. Anand, Z. Li, K. Jun, K. A. Persson, and G. Ceder. “Overcoming systematic softening in universal machine learning interatomic potentials by fine-tuning”. In: (May 2024). URL: https://arxiv.org/pdf/2405.07105.

[84] J. Riebesell, R. E. A. Goodall, P. Benner, Y. Chiang, B. Deng, G. Ceder, M. Asta, A. A. Lee, A. Jain, and K. A. Persson. “Matbench Discovery – A framework to evaluate machine learning crystal stability predictions”. In: *arXiv:2308.14920 [cond-mat.mtrl-sci]* (Aug. 2023), pp. -. URL: https://arxiv.org/pdf/2308.14920.

[85] L. Casillas-Trujillo, A. S. Parackal, R. Armiento, and B. Alling. “Evaluating and improving the predictive accuracy of mixing enthalpies and volumes in disordered alloys from universal pretrained machine learning potentials”. In: *Physical Review Materials* 8.11 (Nov. 2024), p. 113803. ISSN: 24759953. DOI: 10.1103/PHYSREVMATERIALS.8.113803/SUPPLEMENTARY{_}MATERIAL.PDF. URL: https://journals.aps.org/prmaterials/abstract/10.1103/PhysRevMaterials.8.113803.

[86] B. Rhodes, S. Vandenhaute, V. Šimkus, J. Gin, J. Godwin, T. Duignan, and M. Neumann. “Orb-v3: atomistic simulation at scale”. In: (Apr. 2025). URL: https://arxiv.org/pdf/2504.06231.

[87] I. Batatia, S. Batzner, D. P. Kovács, A. Musaelian, G. N. Simm, R. Drautz, C. Ortner, B. Kozinsky, and G. Csányi. "The Design Space of E(3)-Equivariant Atom-Centered Interatomic Potentials". In: *Nature Machine Intelligence* 7.1 (May 2022), pp. 56–67. ISSN: 25225839. DOI: 10.1038/s42256-024-00956-x. URL: https://arxiv.org/pdf/2205.06643.

[88] C. K. Joshi, C. Bodnar, S. V. Mathis, T. Cohen, and P. Liò. "On the Expressive Power of Geometric Graph Neural Networks". In: *Proceedings of Machine Learning Research* 202 (Jan. 2023), pp. 15330–15355. ISSN: 26403498. URL: https://arxiv.org/pdf/2301.09308.

[89] M. Geiger and T. Smidt. "e3nn: Euclidean Neural Networks". In: (July 2022). URL: https://arxiv.org/pdf/2207.09453.

[90] J. A. Nasir, J. Guan, W. Jee, S. M. Woodley, A. A. Sokol, C. Richard, A. Catlow, A.-M. Elena,] Jamal, A. Nasir, and]. C. Richard. "Modelling Silica using MACE-MP-0 Machine Learnt Interatomic Potentials". In: ().

[91] T. Demeyere, T. Ellaby, M. Sarwar, D. Thompsett, and C. K. Skylaris. "Bridging Oxide Thermodynamics and Site-Blocking: A Computational Study of ORR Activity on Platinum Nanoparticles". In: *ACS Catalysis* 15.7 (2025), pp. 5674–5682. DOI: 10.1021/ACSCATAL.5C00321/ASSET/IMAGES/LARGE/CS5C00321{_}0008.JPEG. URL: /doi/pdf/10.1021/acscatal.5c00321?ref=article_openPDF.

[92] Z. W. Ulissi, M. T. Tang, J. Xiao, X. Liu, D. A. Torelli, M. Karamad, K. Cummins, C. Hahn, N. S. Lewis, T. F. Jaramillo, K. Chan, and J. K. Nørskov. "Machine-learning methods enable exhaustive searches for active Bimetallic facets and reveal active site motifs for CO2 reduction". In: *ACS Catalysis* 7.10 (Oct. 2017), pp. 6600–6608. ISSN: 21555435. DOI: 10.1021/ACSCATAL.7B01648/SUPPL{_}FILE/CS7B01648{_}SI{_}001.PDF. URL: https://pubs.acs.org/doi/abs/10.1021/acscatal.7b01648.

[93] D. A. Torelli, S. A. Francis, J. C. Crompton, A. Javier, J. R. Thompson, B. S. Brunschwig, M. P. Soriaga, and N. S. Lewis. "Nickel-Gallium-Catalyzed Electrochemical Reduction of CO2 to Highly Reduced Products at Low Overpotentials". In: *ACS Catalysis* 6.3 (Mar. 2016), pp. 2100–2104. ISSN: 21555435. DOI: 10.1021/ACSCATAL.5B02888/ASSET/IMAGES/LARGE/CS-2015-02888B{_}0004.JPEG. URL: /doi/pdf/10.1021/acscatal.5b02888?ref=article_openPDF.

[94] Y. Pei, T. Z. Gani, Z. Chen, X. Su, Y. Wang, Z. Zhang, K. H. Lim, and S. Kawi. "Catalytic Performance of Ni Catalysts Supported on α-, β-, and γ-Ga2O3 Polymorphs for CO2 Hydrogenation to Methanol". In: *Industrial and Engineering Chemistry Research* 63.17 (May 2024), pp. 7915–7925. ISSN: 15205045. DOI: 10.1021/ACS.IECR.3C03610/ASSET/IMAGES/MEDIUM/IE3C03610{_}M007.GIF. URL: /doi/pdf/10.1021/acs.iecr.3c03610?ref=article_openPDF.

[95] H. Choi, S. Oh, S. B. Trung Tran, and J. Y. Park. "Size-controlled model Ni catalysts on Ga2O3 for CO2 hydrogenation to methanol". In: *Journal of Catalysis* 376 (Aug. 2019), pp. 68–76. ISSN: 0021-9517. DOI: 10.1016/J.JCAT.2019.06.051. URL: https://www.sciencedirect.com/science/article/pii/S0021951719303033.

[96] Y. Shi and F. A. Shakib. "Machine Learning Committee Neural Network Potential Energy Surfaces for Two-Dimensional Metal-Organic Frameworks". In: *Journal of Physical Chemistry C* 129.4 (Jan. 2025), pp. 2222–2230. ISSN: 19327455. DOI: 10.1021/ACS.JPCC.4C07386/ASSET/IMAGES/LARGE/JP4C07386{_}0006.JPEG. URL: /doi/pdf/10.1021/acs.jpcc.4c07386?ref=article_openPDF.

[97] A. Stukowski. “Visualization and analysis of atomistic simulation data with OVITO–the Open Visualization Tool”. In: *Modelling and Simulation in Materials Science and Engineering* 18.1 (Dec. 2009), p. 015012. ISSN: 0965-0393. DOI: 10.1088/0965-0393/18/1/015012. URL: https://iopscience.iop.org/article/10.1088/0965-0393/18/1/015012%20https://iopscience.iop.org/article/10.1088/0965-0393/18/1/015012/meta.

[98] F. Calle-Vallejo. “The ABC of Generalized Coordination Numbers and Their Use as a Descriptor in Electrocatalysis”. In: *Advanced Science* 10.20 (July 2023), p. 2207644. ISSN: 2198-3844. DOI: 10.1002/ADVS.202207644. URL: https://onlinelibrary.wiley.com/doi/full/10.1002/advs.202207644%20https://onlinelibrary.wiley.com/doi/abs/10.1002/advs.202207644%20https://advanced.onlinelibrary.wiley.com/doi/10.1002/advs.202207644.

[99] K. Rossi, G. G. Asara, and F. Baletto. “Structural Screening and Design of Platinum Nanosamples for Oxygen Reduction”. In: *ACS Catalysis* 10.6 (Mar. 2020), pp. 3911–3920. ISSN: 21555435. DOI: 10.1021/ACSCATAL.9B05202/SUPPL{_}FILE/CS9B05202{_}SI{_}001.PDF. URL: https://pubs.acs.org/doi/abs/10.1021/acscatal.9b05202.

[100] Y. Sun, S. Zhou, S. Meng, M. Wang, and H. Mu. “Principal component analysis-artificial neural network-based model for predicting the static strength of seasonally frozen soils”. In: *Scientific Reports* | 13 (123), p. 16085. DOI: 10.1038/s41598-023-43462-7. URL: https://doi.org/10.1038/s41598-023-43462-7.

[101] N. Watanabe, Y. Hori, H. Sugisawa, T. Ida, M. Shoji, and Y. Shigeta. “A machine learning potential construction based on radial distribution function sampling”. In: *Journal of Computational Chemistry* 45.32 (Dec. 2024), pp. 2949–2958. ISSN: 1096-987X. DOI: 10.1002/JCC.27497. URL: /doi/pdf/10.1002/jcc.27497%20https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.27497%20https://onlinelibrary.wiley.com/doi/10.1002/jcc.27497.

[102] E. Bitzek, P. Koskinen, F. Gähler, M. Moseler, and P. Gumbsch. “Structural relaxation made simple”. In: *Physical Review Letters* 97.17 (Oct. 2006), p. 170201. ISSN: 00319007. DOI: 10.1103/PHYSREVLETT.97.170201/FIGURES/2/THUMBNAIL. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.97.170201.

[103] G. C. Cawley and N. L. C. Talbot. “On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation”. In: *Journal of Machine Learning Research* 11 (2010), pp. 2079–2107.

[104] S. Kuhaudomlap, A. Srifa, W. Koo-Amornpattana, C. Fukuhara, and S. Ratchahat. “Insight and comprehensive study of Ni-based catalysts supported on various metal oxides for CO2 methanation”. In: *Scientific Reports 2024 14:1* 14.1 (Oct. 2024), pp. 1–28. ISSN: 2045-2322. DOI: 10.1038/s41598-024-73848-0. URL: https://www.nature.com/articles/s41598-024-73848-0.

[105] W. Liao, C. Tang, H. Zheng, J. Ding, K. Zhang, H. Wang, J. Lu, W. Huang, and Z. Zhang. “Tuning activity and selectivity of CO2 hydrogenation via metal-oxide interfaces over ZnO-supported metal catalysts”. In: *Journal of Catalysis* 407 (Mar. 2022), pp. 126–140. ISSN: 0021-9517. DOI: 10.1016/J.JCAT.2022.01.037. URL: https://www.sciencedirect.com/science/article/pii/S0021951722000379#f0060.

[106] Y. Du, C. Qin, Y. Xu, D. Xu, J. Bai, G. Ma, and M. Ding. “Ni nanoparticles dispersed on oxygen vacancies-rich CeO2 nanoplates for enhanced low-temperature CO2 methanation performance”. In: *Chemical Engineering Journal* 418 (Aug. 2021), p. 129402. ISSN: 1385-8947. DOI: 10.1016/J.CEJ.2021.129402. URL: https://www.sciencedirect.com/science/article/pii/S1385894721009906?via%3Dihub.

[107] R. Barrett, J. C. B. Dietschreit, and J. Westermayr. “Incorporating Long-Range Interactions via the Multipole Expansion into Ground and Excited-State Molecular Simulations”. In: (Feb. 2025). URL: https://arxiv.org/pdf/2502.21045v1.

[108] K. M. Gameel, I. M. Sharafeldin, and N. K. Allam. “First-principles descriptors of CO chemisorption on Ni and Cu surfaces”. In: *Physical Chemistry Chemical Physics* 21.21 (May 2019), pp. 11476–11487. ISSN: 14639076. DOI: 10.1039/C9CP00881K. URL: https://pubs.rsc.org/en/content/articlehtml/2019/cp/c9cp00881k%20https://pubs.rsc.org/en/content/articlelanding/2019/cp/c9cp00881k.

[109] D. Cheng, K. L. C. Nguyen, V. Sumaria, Z. Wei, Z. Zhang, W. Gee, Y. Li, C. G. Morales-Guio, M. Heyde, B. Roldan Cuenya, A. N. Alexandrova, and P. Sautet. “Structure Sensitivity and Catalyst Restructuring for CO2 Electro-reduction on Copper”. In: *Nature Communications* 16.1 (Dec. 2025), pp. 1–14. ISSN: 20411723. DOI: 10.1038/S41467-025-59267-3;TECHMETA=118,119,138,147;SUBJMETA=563,638,639,77,886,979;KWRD=DENSITY+FUNCTIONAL+THEORY,ELECTROCATALYSIS. URL: https://www.nature.com/articles/s41467-025-59267-3.

[110] S. M. Stratton, S. Zhang, and M. M. Montemore. “Addressing complexity in catalyst design: From volcanos and scaling to more sophisticated design strategies”. In: *Surface Science Reports* 78.3 (Aug. 2023), p. 100597. ISSN: 0167-5729. DOI: 10.1016/J.SURFREP.2023.100597.

A

Appendix A: Effective Descriptors for Material Structure in MLIPs

Some criteria must be use for the development of an efficient descriptor in MLIPs. These include Spatial Translation Invariance, which dictates that a descriptor must remain consistent irrespective of shifts in the coordinate system, thereby respecting the isometry of space. Similarly, Rotational Invariance ensures the descriptor is unaltered by coordinate system rotations, adhering to the isotropy principle. Permutation Invariance is another key constraint, meaning the descriptor should be insensitive to the order of atomic indices, reflecting that reordering atoms does not change the system's physical characteristics [62].

Furthermore, Uniqueness is vital, requiring each descriptor to be distinctly tied to a specific atomic environment and property, preventing ambiguous representations. Continuity ensures that the descriptor sensitively reflects minor variations in atomic arrangements, maintaining a proportional relationship between small structural changes and descriptor adjustments. For practical applicability, Compactness is desired, where the descriptor provides comprehensive information while minimizing features to avoid unnecessary complexity. Lastly, Computational Efficiency is paramount, demanding that descriptor calculation be significantly less resource-intensive than direct computations of the physical properties it represents, making MLIPs a viable alternative for large-scale simulations.

Figure A.1 shows an example of commonly used descriptors that appear in different MLIP algorithms.

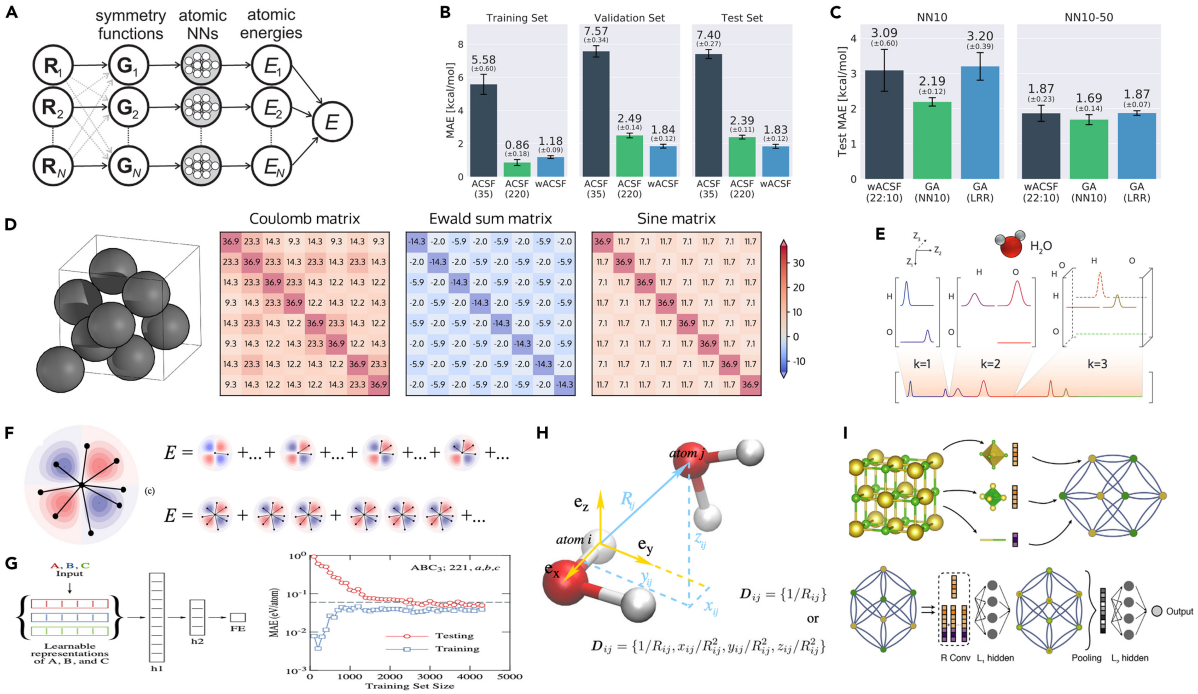


Figure A.1: Some Descriptors seen in MLIPs, as shown in [62]

B

Appendix B: MPNN Interatomic Potentials Equations

Equation B.1 shows how the state of a node is represented, where the position in 3-dimensional space, element composition, and learnable features of the system are located. Equation B.2 shows the construction of a message $m_i^{(t)}$, where one message is created for each node by pooling over its neighbors. The variable M_t represents a learnable message function, and a permutation invariant pooling operation over the neighbors of the node's atom is also applied ($\oplus$), which could be a mathematical operation such as a sum. In the update step, the message is transformed into new features. U_t here represents a learnable update function, and after a specified number of message constructions (here represented as T) and update steps, a readout function (R) is applied to map the node states to the target property that would be calculated, for example, the site energy of the atom (E_i).

$$\sigma_i^{(t)} = (\mathbf{r}_i, z_i, \mathbf{h}_i^{(t)}) \tag{B.1}$$

$$\mathbf{m}_i^{(t)} = \oplus_{j \in N(i)} M_t(\sigma_i^{(t)}, \sigma_j^{(t)}) \tag{B.2}$$

$$h_i^{(t+1)} = U_t(\sigma_i^{(t)}, \mathbf{m}_i^{(t)}) \tag{B.3}$$

$$E_i = \sum_{t=1}^{T} R_t(\sigma_i^{(t)}) \tag{B.4}$$

C

Appendix C: INCAR file setup

To initialize a VASP setup for energy minimization, single point calculations, or molecular dynamics, the input parameters must be correctly defined. In this research project, VASP was used to do at least one run for each of the aforementioned simulations. Figure C.1 shows a screenshot of an INCAR energy minimization setup. The parameters and a short description of each [41] listed here are the following:

- **SYSTEM**: User given name of the simulation or test to be run.
- **NWRITE**: Verbosity tag for the OUTCAR file. Determines how much will be written in the OUTCAR. By default it equals 2, where information such as eigenvalues, DOS+charge density, energies, forces, and stress will be included.
- **PREC**: Precision mode, which sets default values for the energy cutoff.
- **ISTART**: Job begins from scratch when this value is equal to 0. When set to another value, the job will continue according to the orbitals read from a WAVECAR value.
- **ICHARG**: Determines how VASP constructs the initial charge density.
- **ISPIN**: Specifies spin polarization
- **IVDW**: Specifies a Van der Waals dispersion term of atom-pairwise or many-body type.
- **MAGNOM**: Initial magnetic moment for each atom if no magnetization density is present.
- **ENCUT**: Specifies energy cutoff for the plane-wave basis set in eV.
- **EDIFF**: Specifies the global break condition for the electronic SC-loop.
- **LREAL**: Determines if projection operators are evaluated in real-space or in reciprocal space.
- **NELMIN**: Determines minimum number of electronic self-consistency steps.
- **ALGO**: Specifies the electronic minimization algorithm or selects the type of GW calculations.
- **ISIF**: Determines if the stress tensor is calculated and its degrees of freedom.
- **NSW**: Sets the maximum number of ionic steps. Set to 1 for single point calculations.
- **IBRION**: Determines how the crystal structure changes during the calculation. 0 is for MD, 2 for conjugate gradient structure optimization.
- **POTIM**: Sets the time step in MD or step width in ionic relaxations.
- **ISMEAR**: Determines how the partial occupancies are set for each orbital.
- **SIGMA**: Determines the width of the smearing in eV.

- **NPAR**: Number of bands that are treated in parallel.
- **ISYM**: Determines how VASP treats symmetry. When = 0, symmetry is switched off.

```
   SYSTEM   =  opt_2

# Start parameter for this run:
    NWRITE =      2                           # write-flag & timer
    PREC   =      normal                      # normal, accurate
    ISTART =      0                           # job   : 0-new  1-cont  2-samecut
    ICHARG =      2                           # charge: 1-file 2-atom 10-const
    ISPIN  =      2
    IVDW   =     12
    MAGMOM = 160*0.0 8*1.0
# Electronic Relaxation 1
    ENCUT = 400
    EDIFF   =  0.1E-04                        # stopping-criterion for ELM
    LREAL   =  Auto                           # real-space projection
    NELMIN  =  8



# Electronic relaxation 2 (details)
    ALGO   =   Fast

# Ionic relaxation
    EDIFFG  =   -0.06                         # stopping-criterion for IOM, negative-force  positive-energy
    ISIF    =    2                           # stress and relaxation
    NSW     =     300                        # number of steps for IOM
    IBRION  =      2                          # ionic relax: 0-MD 1-quasi-New 2-CG
    POTIM = 0.25
    !IOPT = 1

# DOS related values
    ISMEAR =     0;
    SIGMA  =   0.05                           # broadening in eV -4-tet -1-fermi 0-gaus


# Write flags
    LWAVE   =      F                          # write WAVECAR
    LCHARG  =      F                          # write CHGCAR
    LVTOT   =      F                          # write LOCPOT, local potential
    NPAR    =      16

ISYM = 0
```

Figure C.1: INCAR file setup example for energy minimization

For the MD simulation setup, there are additional parameters that have been setup. These are the following:

- **TEBEG**: Sets the initial temperature in Kelvin of the MD simulation.
- **TEEND**: Sets the final temperature in Kelvin of the MD simulation.
- **SMASS**: Controls the velocities during an ab-initio MD run. -1 to setup a Langevin thermostat.

```
# Molecular Dynamics Settings for NVT
    IBRION  = 0           # MD mode
    TEBEG   = 323.15      # Initial temperature (K)
    TEEND   = 323.15      # Final temperature (K)
    SMASS   = -1          # Langevin thermostat
    POTIM   = 1.0         # MD time step in fs
    NSW     = 500        # Number of MD steps (adjust based on desired simulation length)
    ISIF    = 0           # No cell relaxation
```

Figure C.2: INCAR file setup example for NVT ensemble MD

D

Appendix D: Configurations of Ni/Ga_2O_3 with Hydrogen Atoms before and after finetuning

The first active learning loop was done using Ni/Ga_2O_3 with hydrogen atoms adsorbed on the nickel cluster and also on the Ga_2O_3 support (see fig D.1). This data is also part of the same database given by the Inorganic Systems Engineering group. Throughout the active learning loop, molecular dynamics simulations were ran to observe how well the finetuned foundational model calculates energies and forces in these simulations as of compared to the DFT based simulations. In this case, the nickel atoms tend to cluster one on top of each other, where only two nickel atoms formed bonds with the oxygen atoms located in the support. This lack of oxygen bridges, low dispersion and surface area of the nickel cluster, and a clear inappropriate interaction between the metal and support reflects the poor performance of this model [104, 105].

Later on, it was found that this behavior is due to the lack of the multihead replay feature utilized in finetuning. With this feature turned on, the model's generalization capabilities are kept. The finetuning for this active learning included many elements and configurations (so the finetuning dataset was not filtered correctly, since it included information from other metal structures found in the MP0 database). This can also lead to an undesired behavior on this catalyst, since the Materials Project Database contains configurational information of metallic crystals composed of one or more metals. These may also include non-metallic elements such as O and H, however, the formation of intermetallic bonding is enhanced due to the metal-metal bonding seen in many of the Materials project database, which could lead to this nickel cluster formation [76].

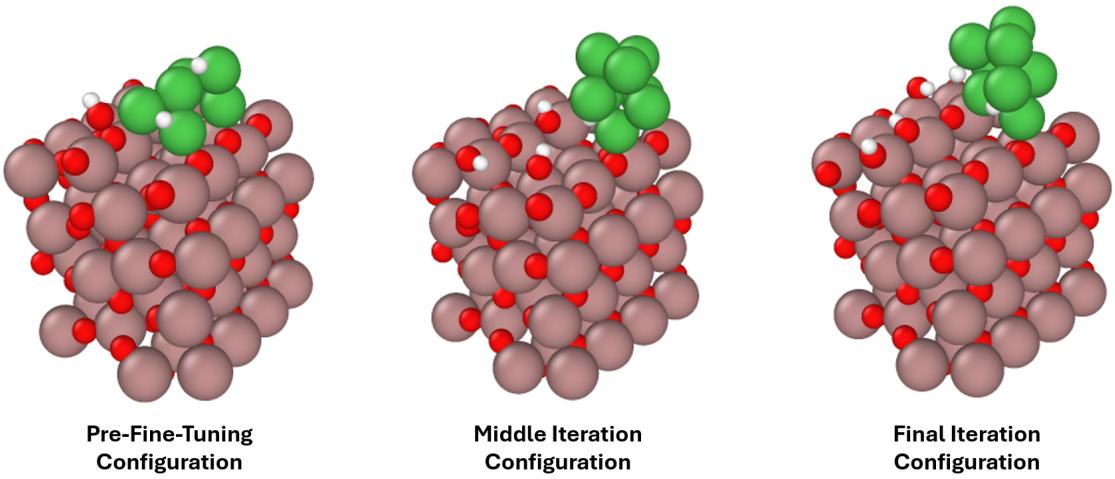


Figure D.1: Ni/Ga_2O_3 with Hydrogen configurations evolutions throughout active learning loop

Appendix E: Single PCA analyses of MP-0 based Active Learning Loops

All PCAs for the MP-0 based MD simulation are presented here in figures E.1-E.3. Committee based PCAs are seen in figure E.4. These individual PCA analyses confirm that at the beginning of the molecular dynamics trajectory, the structures gradually transition to more stable configurations.

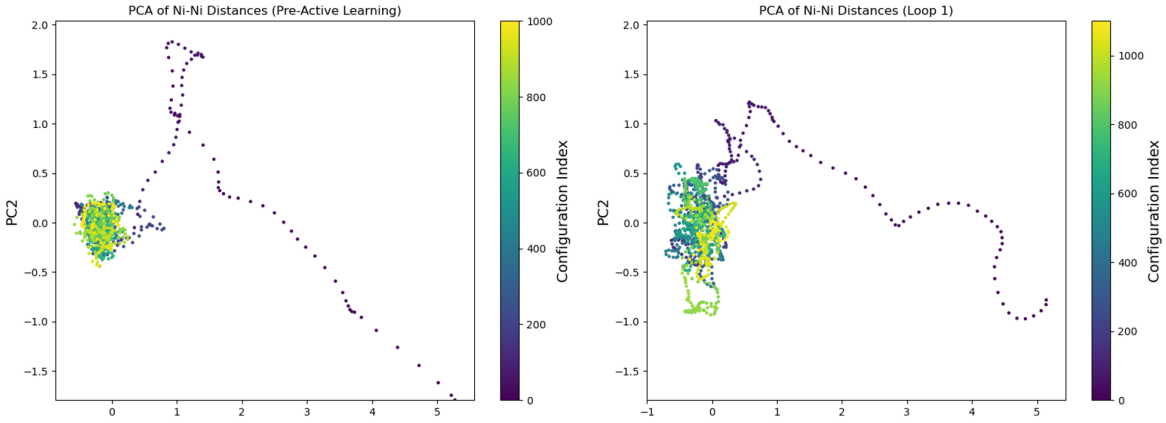


Figure E.1: PCA pre-finetuning and loop 1 MP0 based MD

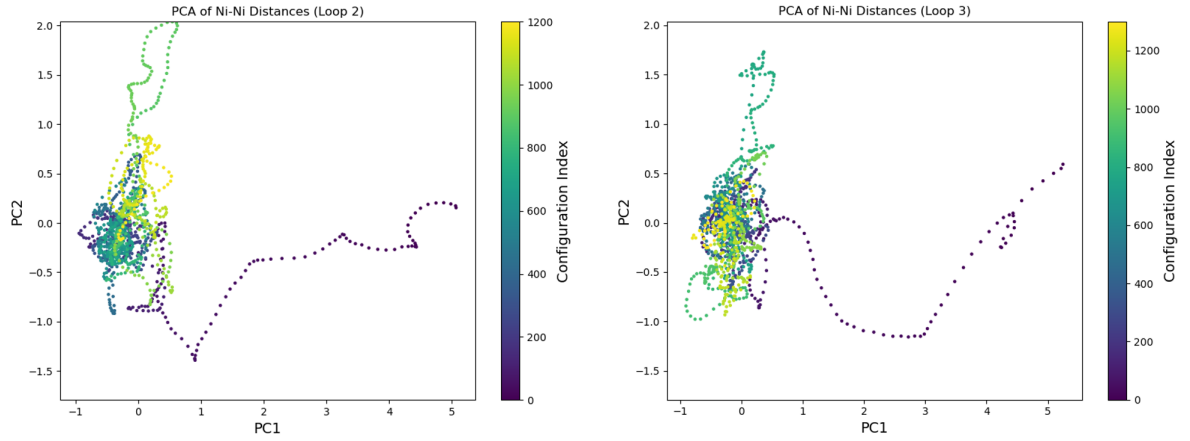


Figure E.2: PCA loop 2 and 3 MP0 based MD

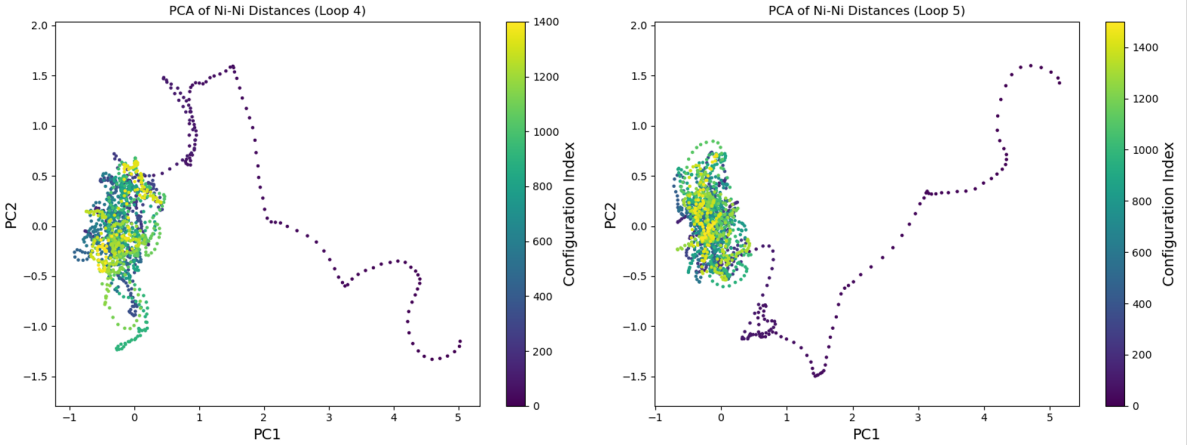


Figure E.3: PCA loop 4 and 5 MP0 based MD

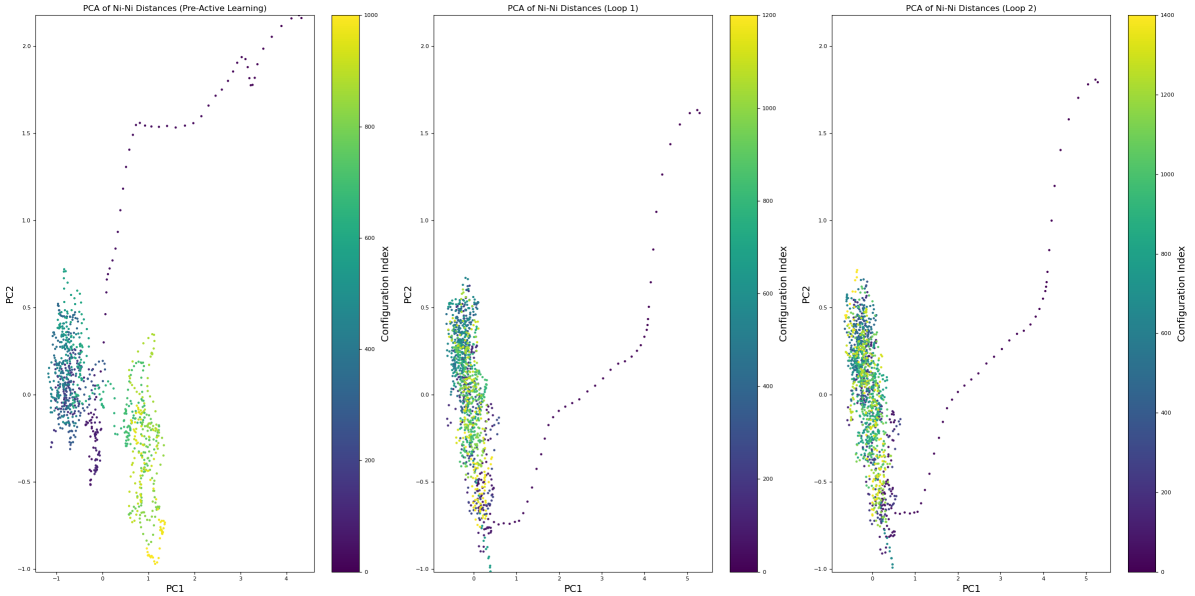


Figure E.4: PCA over all active learning loops MACE-Committee

Appendix F: Energies and Bonding Distances Plots for MP0 MD Simulation

Here, all of the kinetic, potential, and total energies for the different foundational model based MD simulations are shown in figures F.1-F.9. Pair distance distribution between all possible elemental bonds found in the Ni/Ga_2O_3 system are shown from figures F.10-F.27. These are examples of the automatically outputted plots obtained per each iteration of the active learning loop. These graphs are useful to the user if they are interested in observing the dynamics behavior evolution over every iteration of the loop, where directly the energies, coordination, or nearest neighbor distance of specific bonds want to be studied.

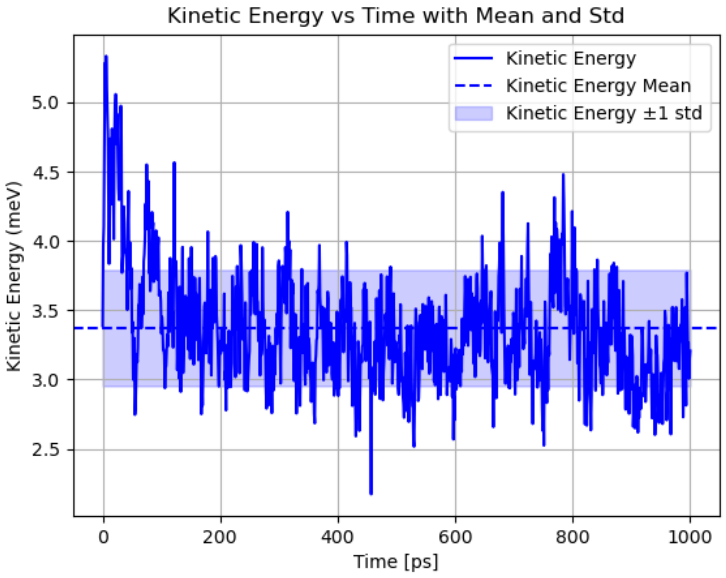


Figure F.1: Pre-Fine tuning, Kinetic Energy

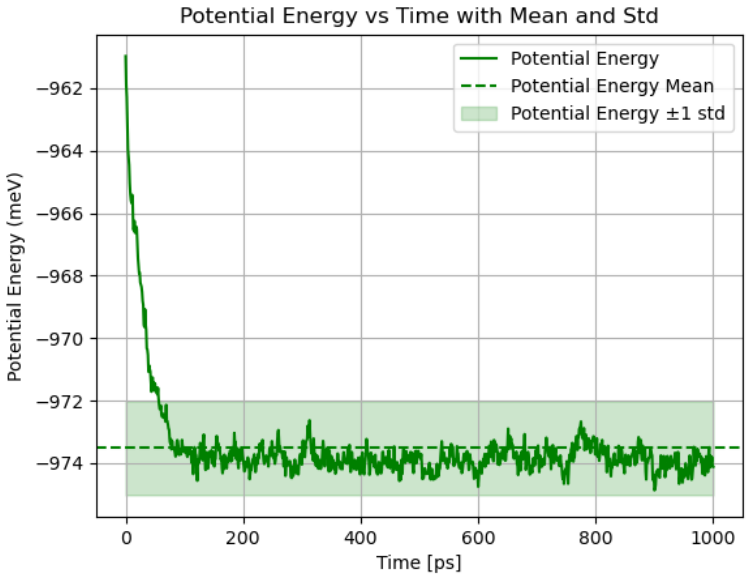


Figure F.2: Pre-Fine tuning, Potential Energy

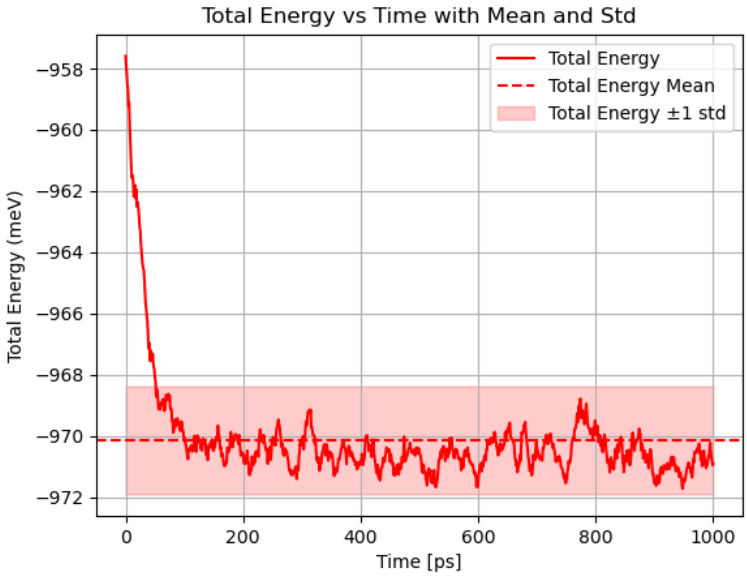


Figure F.3: Pre-Fine tuning, Total Energy

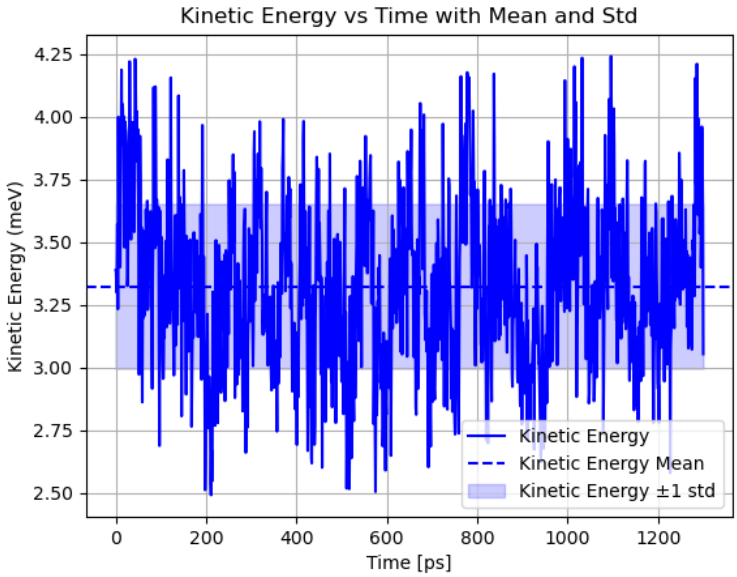


Figure F.4: Iteration 3, Kinetic Energy

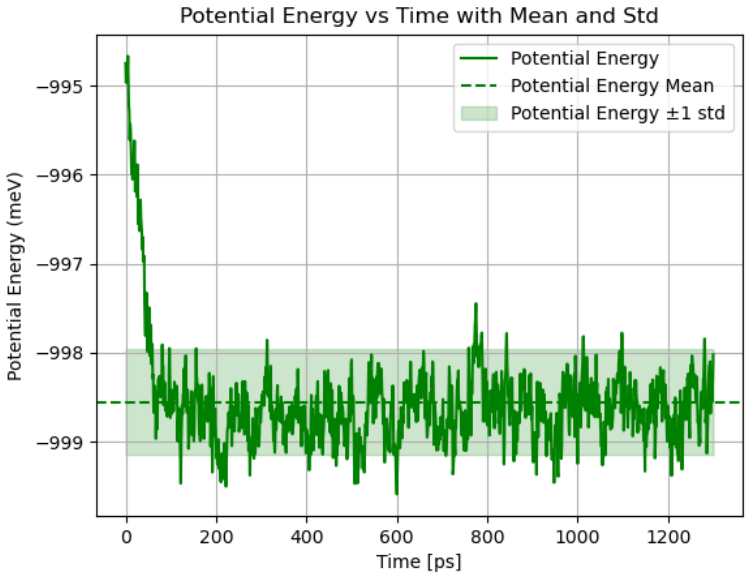


Figure F.5: Iteration 3, Potential Energy

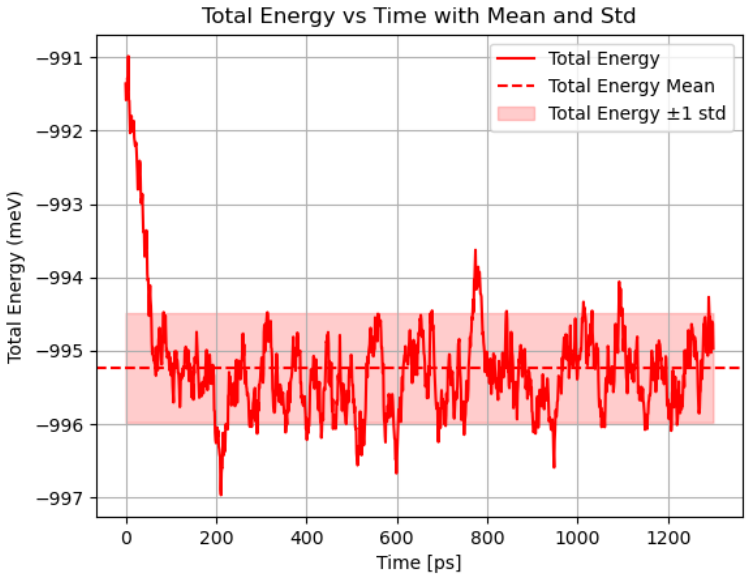


Figure F.6: Iteration 3, Total Energy

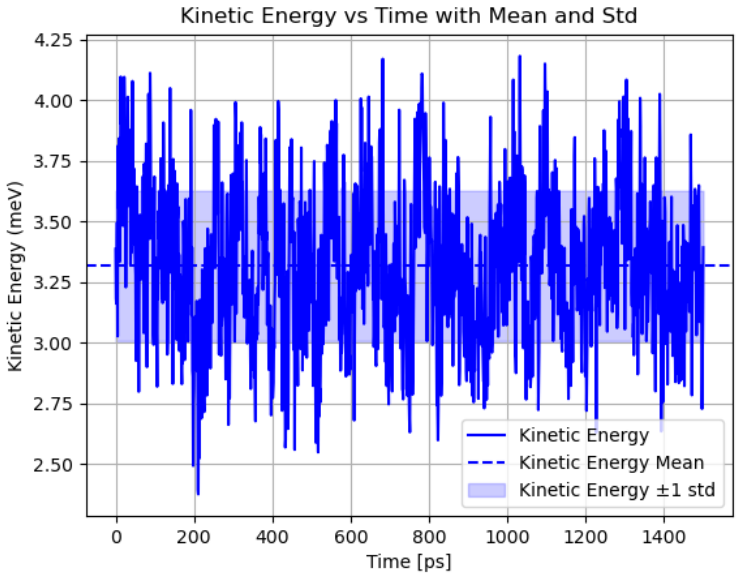


Figure F.7: Iteration 5, Kinetic Energy

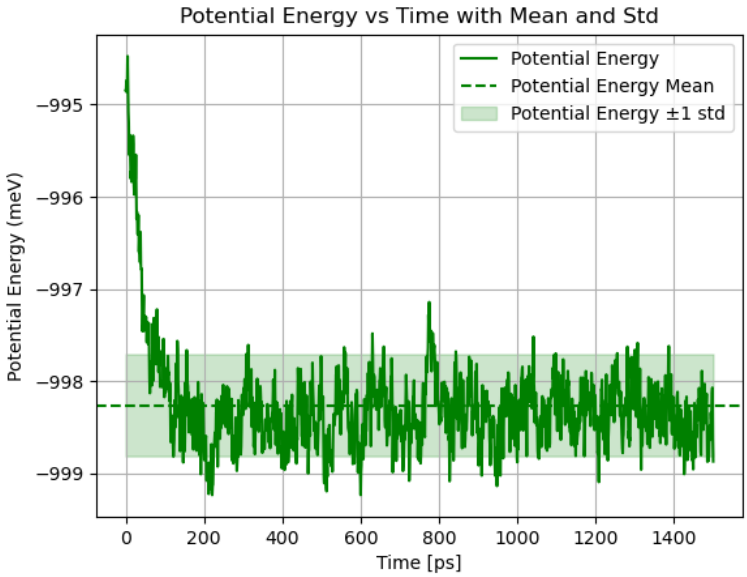


Figure F.8: Iteration 5, Potential Energy

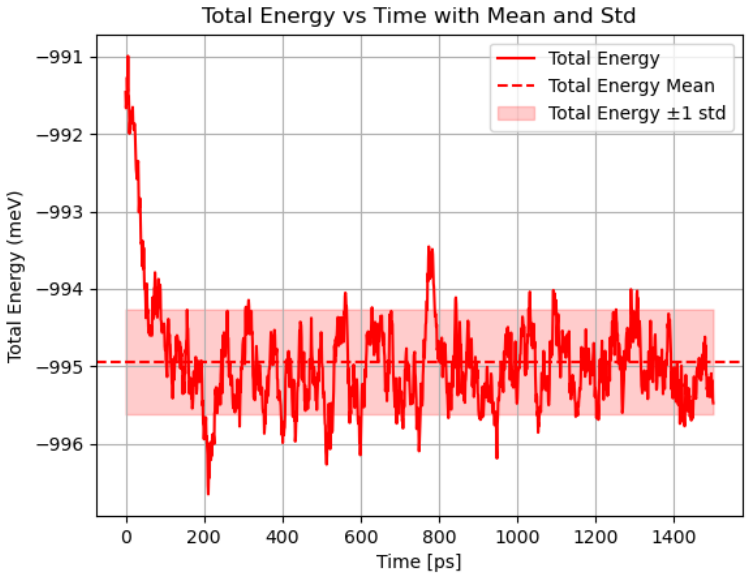


Figure F.9: Iteration 5, Total Energy

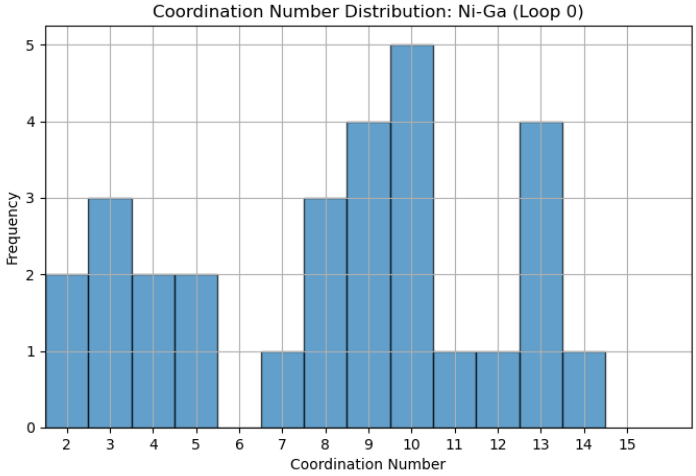


Figure F.10: Coordination Number Distribution of Ni-Ga bonds, Pre-finetuning

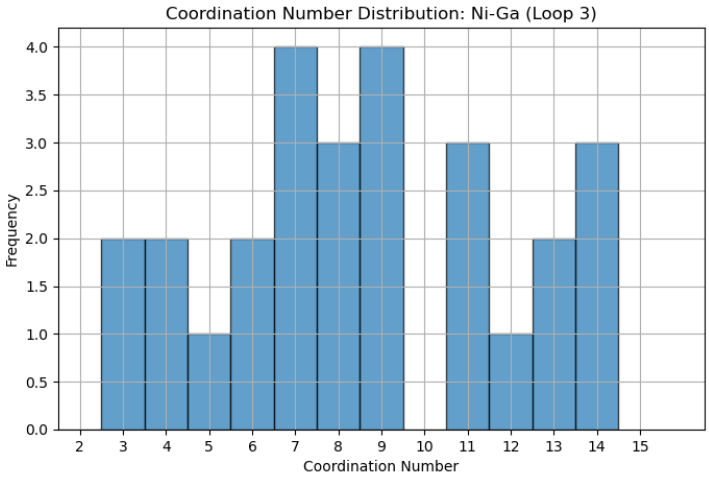


Figure F.11: Coordination Number Distribution of Ni-Ga bonds, Loop 3

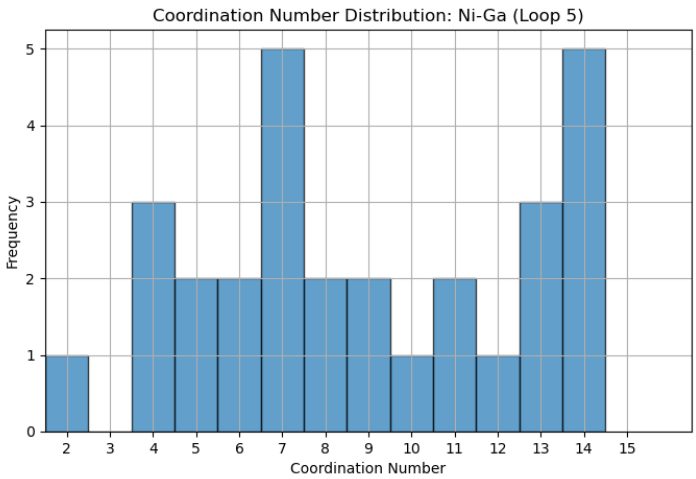


Figure F.12: Coordination Number Distribution of Ni-Ga bonds, Loop 5

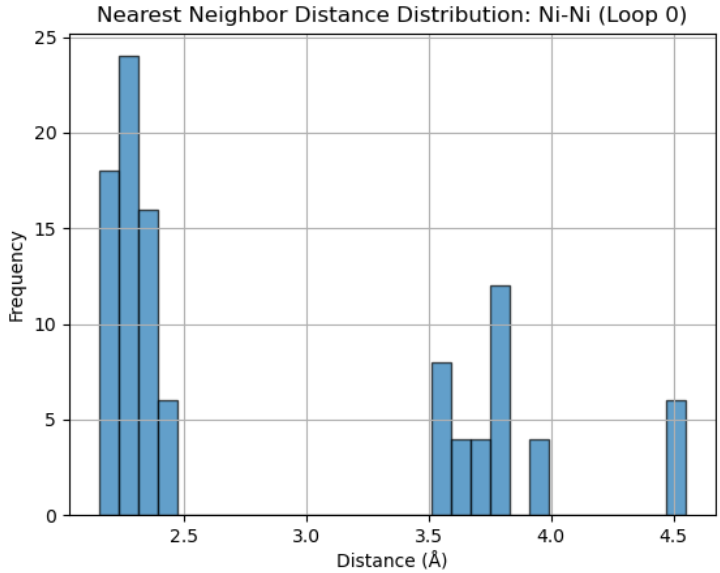


Figure F.13: Nickel-Nickel bond distribution, Pre-finetuning

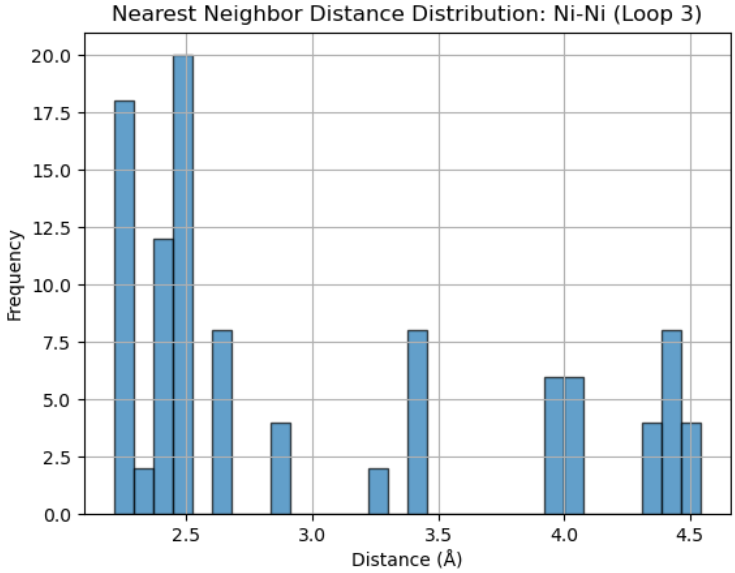


Figure F.14: Nickel-Nickel bond distribution, Loop 3

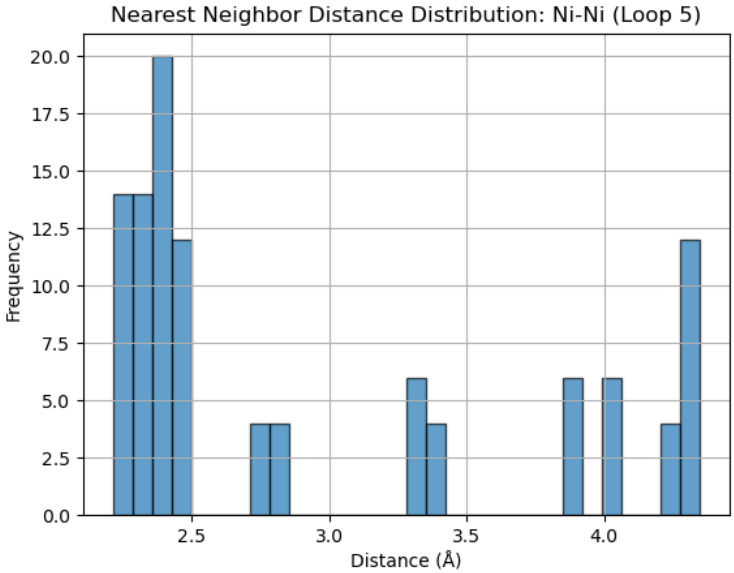


Figure F.15: Nickel-Nickel bond distribution, Loop 5

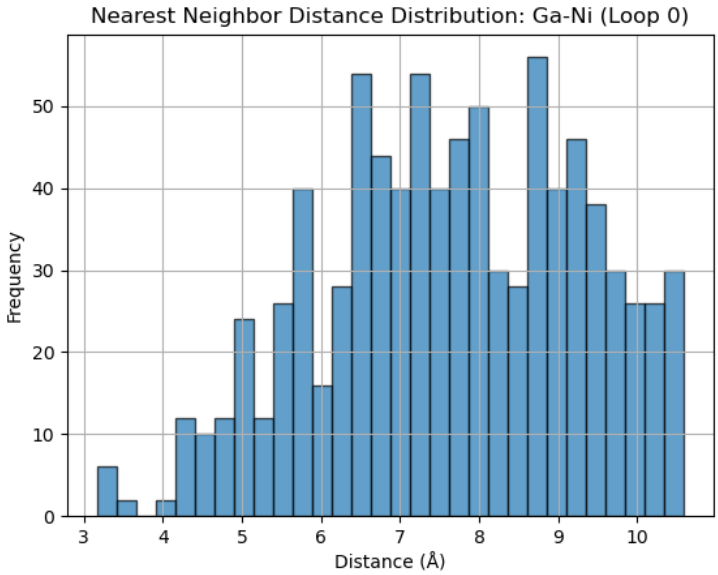


Figure F.16: Nickel-Gallium bond distribution, Pre-finetuning

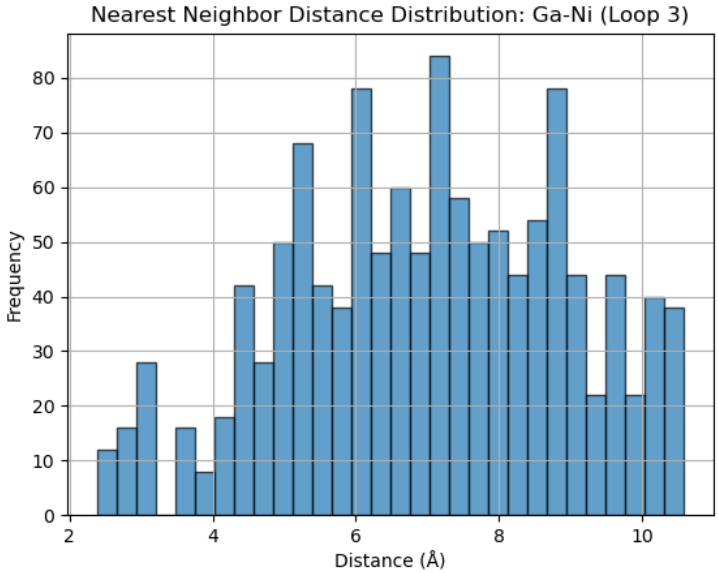


Figure F.17: Nickel-Gallium bond distribution, Loop 3

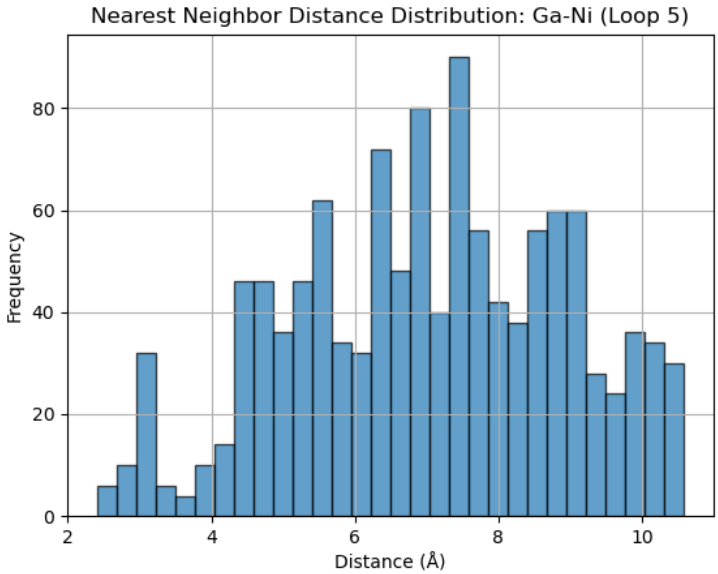


Figure F.18: Nickel-Gallium bond distribution, Loop 5

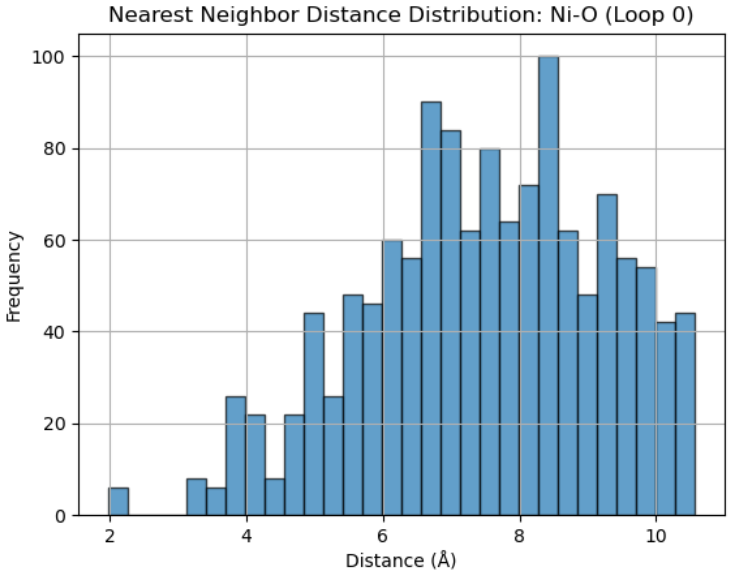


Figure F.19: Nickel-Oxygen bond distribution, Pre-finetuning

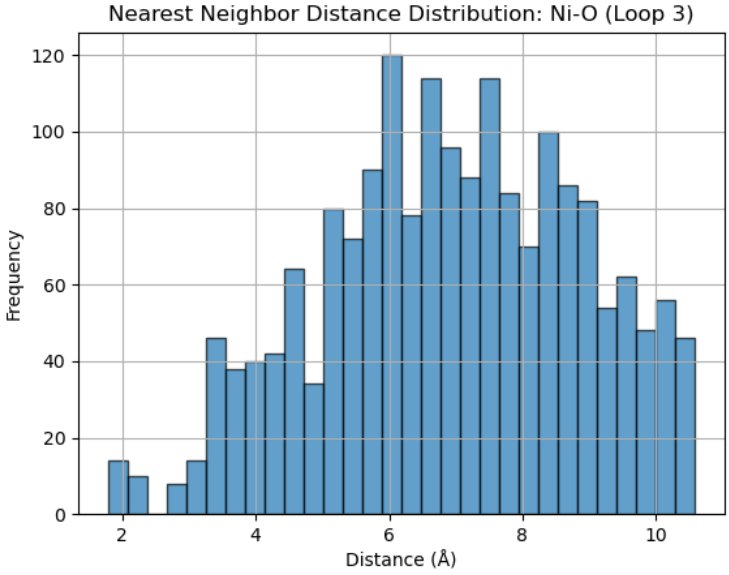


Figure F.20: Nickel-Oxygen bond distribution, Loop 3

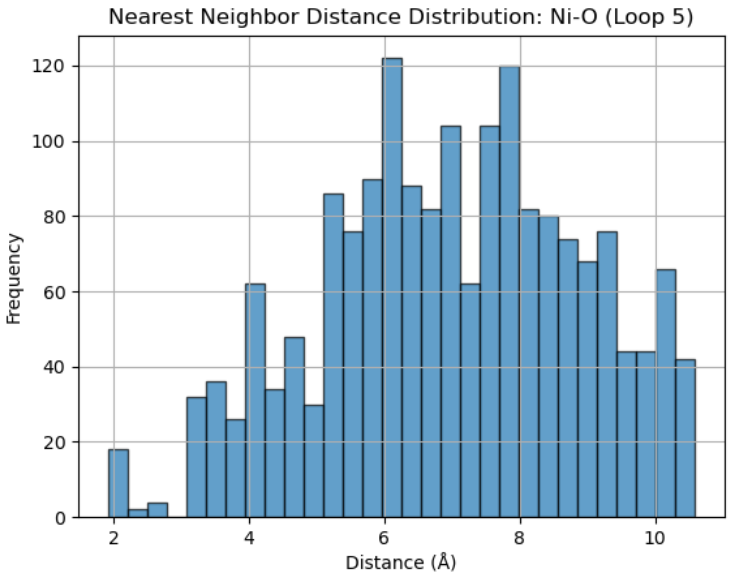


Figure F.21: Nickel-Oxygen bond distribution, Loop 5

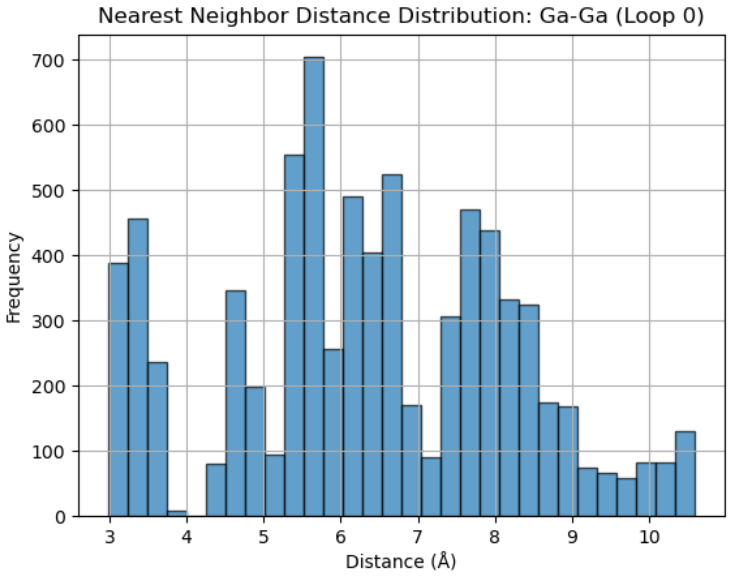


Figure F.22: Gallium-Gallium bond distribution, Pre-finetuning

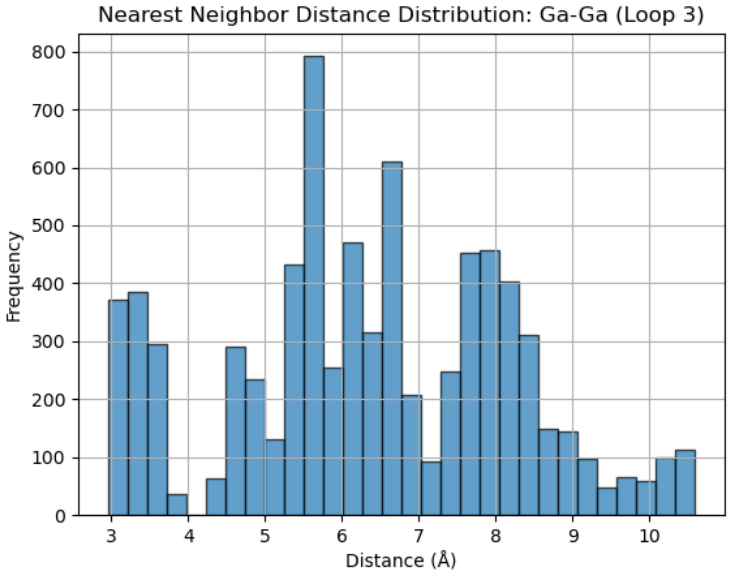


Figure F.23: Gallium-Gallium bond distribution, Loop 3

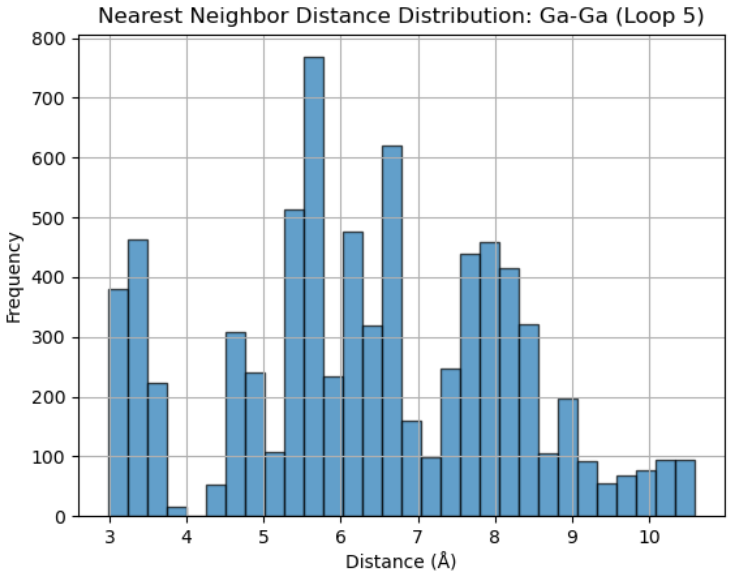


Figure F.24: Gallium-Gallium bond distribution, Loop 5

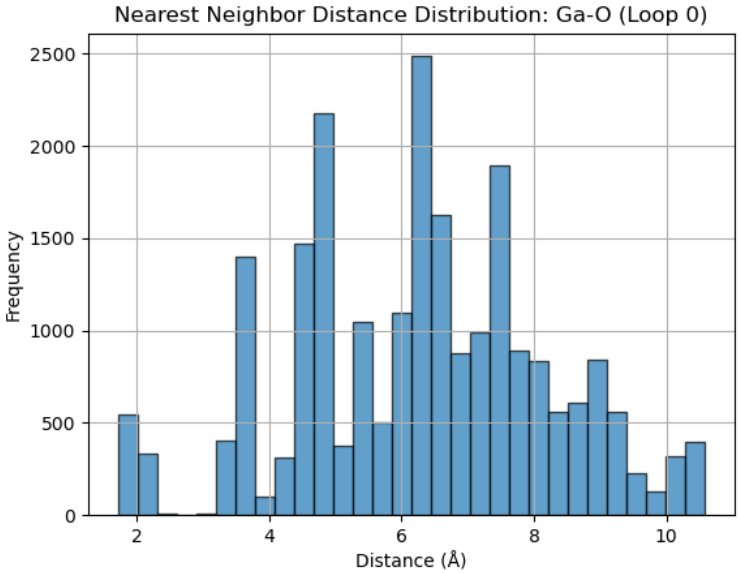


Figure F.25: Gallium-Oxygen bond distribution, Pre-finetuning

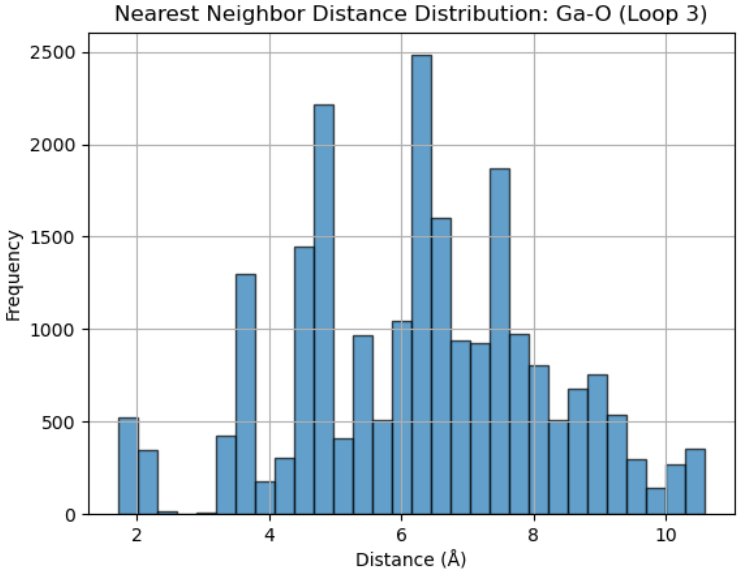


Figure F.26: Gallium-Oxygen bond distribution, Loop 3

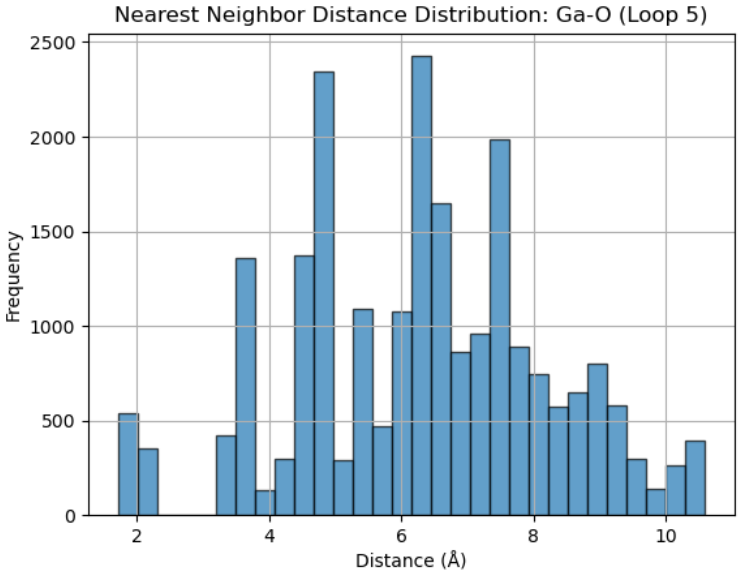


Figure F.27: Gallium-Oxygen bond distribution, Loop 5