Spring 2016

Molecular Dynamics Simulations of a Pentacene Derivative on Gold Surfaces in the Quantum Mechanical and Classical Mechanics Frameworks

Ryan Allen Miller

University of New Hampshire, Durham

Follow this and additional works at: https://scholars.unh.edu/thesis

Recommended Citation


https://scholars.unh.edu/thesis/1087

This Thesis is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Master's Theses and Capstones by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact nicole.hentz@unh.edu.
Molecular Dynamics Simulations of a Pentacene Derivative on Gold Surfaces in the Quantum Mechanical and Classical Mechanics Frameworks

BY

RYAN MILLER

B.A. in Physics, University of California, Berkeley, 2014

THESIS

Submitted to the University of New Hampshire
in partial fulfillment of
the requirements for the degree of

Master of Science

in

Physics

May, 2016
This thesis has been examined and approved in partial fulfillment of the requirements for the degree of Master of Science in Physics by:

Thesis Director, Karsten Pohl
Professor of Physics and Materials Science

Glen P. Miller
Chair of Chemistry Department
Professor of Chemistry and Materials Science

David Mattingly
Assistant Professor of Physics

On May 02, 2016

Original approval signatures are on file with the University of New Hampshire Graduate School.
DEDICATION

This work is dedicated to my grandma Adele, for always fostering my scientific interest and being a pillar of perseverance and knowledge.
ACKNOWLEDGMENTS

The completion of this research would not have been possible without the company of the wonderful people I have in my life, and the fantastic guidance I received throughout this work. First and foremost, I thank Dr. Amanda Larson for being not only a fantastic mentor and an excellent researcher to learn from, but also being very kind, patient, and helpful throughout this project. From day one Amanda did everything necessary to make sure I understood what was going on, and helped me transition into my role in the research group. This work certainly would not have been possible without her guidance and mentorship, and I am very grateful to have had her to lean on for support.

I thank my friends that I made here in New Hampshire for being so kind and enjoyable to be surrounded by. While there are many, I send special thanks to David for becoming such a close and supportive friend to me and not being afraid to spot me in the squat rack day in and day out, and to Jake for being truly the best roommate I could have asked for the last two years.

I would like to thank other members of the Pohl group: Karsten, for superb guidance as my research advisor, and Maxwell Grady, Zhongwei Dai, and (now Dr.) John McClain for being both helpful to me and also great friends. I thank Dr. Glen Miller and Dr. David Mattingly for serving on my thesis committee and for their guidance.

Finally, I thank both my family and my wonderful girlfriend Hillary and her family. I truly couldn’t have asked to be part of better families, and having their support certainly played an invaluable role in being able to complete this work.
# Table of Contents

**Dedication** ................................................................. iii

**Acknowledgments** ......................................................... iv

**List of Tables** .......................................................... ix

**List of Figures** .......................................................... x

**Abstract** ........................................................................... xv

1 **Introduction and Motivation** ........................................... 1

1.1 Organic Photovoltaic Devices ........................................... 1

1.2 Pentacene and TTPO ...................................................... 4

1.2.1 Pentacene ............................................................... 4

1.2.2 TTPO ................................................................. 5

Thermal Stability ................................................................. 6

Crystallization and Solubility .................................................. 6

Synthesis Procedure ............................................................. 7

1.3 Molecular Dynamics Simulations ...................................... 7

2 **Molecular Dynamics Theory** ............................................ 9

2.1 First-Principles Methods ................................................ 9

2.1.1 The Electron Density ............................................... 9

2.1.2 The Hohenberg-Kohn Theorems ................................... 10

2.1.3 The Kohn-Sham Equations ......................................... 12
Local Density Approximation ........................................... 13
2.1.4 The Self Consistent Field Loop ................................. 14
2.1.5 Born-Oppenheimer Molecular Dynamics ..................... 16
2.1.6 Car-Parrinello Molecular Dynamics ......................... 18
2.2 Classical MD ............................................................ 21
2.3 Integrating the Equations of Motion ............................. 23
2.3.1 The Verlet Algorithm ............................................ 23
2.4 Thermostat Background ............................................ 24
2.4.1 The Langevin Equation and Brownian Motion ............... 24
2.4.2 Nosé-Hoover Thermostat ....................................... 26

3 Computational Specifications and Methods ....................... 30
3.1 Molecular Dynamics in Quantum ESPRESSO .................. 30
3.1.1 Executables ....................................................... 30
PWscf (pw.x) ............................................................ 31
Input Parameters for PWscf ........................................... 32
cp.x ................................................................. 35
3.1.2 Periodicity ....................................................... 36
The Plane Wave Basis ................................................... 36
The Unit Cell ........................................................... 37
3.1.3 K-Point Sampling and Ultrasoft Pseudopotentials .......... 39
3.1.4 Temperature Control ........................................... 41
3.2 LAMMPS ............................................................ 42
3.2.1 Harmonic Bonds ............................................... 44
3.2.2 Angular Potential Energy ........................................... 45
3.2.3 Dihedrals (Torsional Energy) ................................... 45
3.2.4 Potentials (Pair Styles) ........................................... 46

3.3 Visualization and Input/Data File Generation .................... 52
3.3.1 PWgui ............................................................... 52
3.3.2 XCrysDen ......................................................... 52
3.3.3 VMD ................................................................. 54
3.3.4 TopoTools .......................................................... 54
3.3.5 Moltemplate ......................................................... 55

3.4 Machines ............................................................... 57
3.4.1 Vortex ............................................................... 57
SLURM ................................................................. 57
3.4.2 Trillian ............................................................... 58
PBS ................................................................. 58

4 THE GOLD SUBSTRATE AND SOLID-STATE SURFACE DIFFUSION 60

4.1 π-Conjugation .......................................................... 60
4.1.1 Angular Assembly ................................................. 61

4.2 Solid-State Surface Diffusivity ...................................... 62
4.2.1 Diffusion Coefficient from Langevin Dynamics ............... 64
4.2.2 Arrhenius Plots and Diffusion Barrier Energy ............... 67
4.2.3 Substrate Dependence and Au(111) Reconstruction .......... 69
Vacancy defects ......................................................... 71
High-Index Cuts: Vicinal Surfaces ..................................... 72
5 Results

5.1 Comparison of DFT and Classical Mechanics Results .................................. 75
  5.1.1 Tilt Angle ................................................. 75
  5.1.2 Close-Packed Direction Alignment ................................................. 82
5.2 Stable Chain Formation at Step Edges at 373K ............................................ 85
5.3 Partial Monolayer Coverage on a Flat Surface .............................................. 87
  5.3.1 Cluster Formation ............................................. 87
  5.3.2 Activation Energy for Diffusion ................................................. 90
5.4 Site Vacancies .......................................................... 96
  5.4.1 Substrate Trenches: The Lock-and-Key Effect ..................................... 98

6 Conclusions .................................................................................. 102
  6.1 Overview and Summary .............................................................. 102
  6.2 Possible Future Work ............................................................... 105

Appendices .................................................................................. 107

Appendix A Preparing a LAMMPS Data File .............................................. 108
Appendix B Example LAMMPS Input File ................................................. 111
Appendix C Installing and Building LAMMPS on Trillian ............................... 114

Bibliography .................................................................................. 118
# List of Tables

3.1 Extra Variables to be Set for Relax Calculation ...................... 34
3.2 Extra Lines for Dipole Correction ..................................... 38
3.3 Example of Variable Input in LAMMPS .............................. 43
3.4 Interaction Parameters for LAMMPS Simulations ...................... 51
5.1 Diffusion Coefficients for TTPO on Flat Surface ...................... 94
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-1</td>
<td>NCPV solar cell efficiency data. Note the interesting steep increase in efficiency for organic-material based solar cells (red filled-in circles).</td>
<td>2</td>
</tr>
<tr>
<td>1-2</td>
<td>Schematic showing the basic idea of a bilayer OPV. The donor molecule could be a pentacene derivative like the one analysed in this project, the acceptor is usually C$_{60}$ or a derivative, while the other materials are chosen accordingly. From [1].</td>
<td>3</td>
</tr>
<tr>
<td>1-3</td>
<td>Molecular structure of TTPO</td>
<td>5</td>
</tr>
<tr>
<td>2-1</td>
<td>A visual showing one iteration of the SCF process.</td>
<td>15</td>
</tr>
<tr>
<td>2-2</td>
<td>A visual showing the outer loop for ionic relaxation in BOMD with the SCF loop from above.</td>
<td>18</td>
</tr>
<tr>
<td>3-1</td>
<td>Example input file for an SCF calculation. Most atomic coordinates have been suppressed for brevity.</td>
<td>32</td>
</tr>
<tr>
<td>3-2</td>
<td>Two dimensional BZ showing $E_{cut}$</td>
<td>40</td>
</tr>
<tr>
<td>3-3</td>
<td>Figure showing a molecular dihedral. Displacing an end molecule into or out of the page results in a torsion about the bond between the middle two atoms, and a resulting harmonic potential energy. From [2].</td>
<td>46</td>
</tr>
<tr>
<td>3-4</td>
<td>A sketch of the Morse potential overlayed with a traditional harmonic (symmetric) potential with equally spaced energy intervals, where x is the inter-atomic spacing. From the National Institute of Standards and Technology [3].</td>
<td>48</td>
</tr>
</tbody>
</table>
3-5 Partial charges used for TTPO molecule (in fractions of e). The Natural Bond Orbital (NBO) analysis was performed in [4] for a two-molecule columnar stack in the DFT framework. ................................................. 50

3-6 Example of a batch submission script in SLURM. ................................................................. 57

3-7 Example of a batch submission script in PBS. ................................................................. 59

4-1 Two possible reconstruction patterns for planar organic molecules. (a) shows herringbone reconstruction, which is the observed pattern for pentacene. (b) shows lamellar stacking, the preferred arrangement for charge-carrying applications. From [5]. .................................................. 61

4-2 Pentacene and two of its derivatives isolated on Au(111). TTPO is unique because the sulfur end tilts towards the surface. ........................................... 62

4-3 Consecutive STM images of TTPO on Au(788) taken at 160 seconds apart. The blue and green arrows remain at the exact same location in each image. Focusing on one color arrow at a time, and moving top to bottom, left to right, one can see that the molecules do not remain stationary, despite no external intervention. From [6]. .................................................. 63

4-4 Simplified graphic of the potential energy surface of a metal surface. In (a), two adatoms are shown in a stable state (position 1, in a bridge or well site), and a metastable state (position 2, on top of a surface atom). (b) shows a theoretical map of potential energy of a single well, which is modelled as an asymmetric well. Finally, part (c) demonstrates the concatenation of several potential wells which models the metal surface and the diffusion landscape. Adapted from [7]. .................................................. 68

4-5 Different representations of fcc (111) surface and close-packed formations. From [8]. .................................................. 70

4-6 STM images showing herringbone reconstruction characteristics. From [9]. .... 71
4-7 Two types of stable vicinal Au(111) surfaces. On the left, the steps are \{111\}-like (triangular packing). On the right, they are \{100\}-like (square packing). The terraces retain the 111 pattern. From [10].

4-8 High-resolution STM image showing the ordered chain structure of TTPO on Au(788) steps. From [6].

5-1 DFT relax results. Periodic boundary conditions were enforced in all three directions, and the kinetic energy cutoff for wavefunctions was 30.0 Ry.

5-2 Classical mechanics relax results. Physical environment and initial configuration were the same as for DFT.

5-3 DFT relax results for one molecule at the step edge, performed in [6]. Angles of minimum energy are shown for the step-facing and facing-away configurations.

5-4 LAMMPS relax results for one molecule at the step edge. Angles for the facing-away and step-facing assemblies with respect to the solid line are 9.3° and 12.9°, respectively.

5-5 Small comparative molecular dynamics simulation ran in the DFT and classical-mechanics based frameworks.

5-6 A small DFT simulation showing a clear preference for alignment of the sulfur bridge along a close-packed direction. (a) The molecule lowered itself to the surface from the initial placement and raised its temperature to 300K. (b) The molecule has clearly rotated 60° clockwise to align the sulfur bridge along a close-packed direction, indicated by the red arrow. (c) The molecule moved about one lattice site along the direction of the blue arrow, which is the plane associated with the close-packed symmetry direction. The other molecule can be seen to begin rotating towards the direction as well.
LAMMPS simulation at 300K showing an isolated TTPO molecule travelling along a close-packed symmetry direction. The molecule travelled approximately two lattice sites (twice as far than in the above DFT simulation), and it maintained its alignment in the symmetry direction. The blue arrow in (b) indicates the direction of motion along the symmetry plane, and the red lines show the sixfold symmetry for reference.

Chain formation at an Au(788) step site at 373K. (a) Shows the initial configuration of molecules, and (b) shows the definite alignment of some of the molecules along a step edge and chain formation. (c) Is an adapted STM image from [6] that shows the similarities with the results obtained here.

The three formations found in clusters of TTPO molecules on gold. Clusters can be found in any of the three states, or combinations of the three. The most stable state is the chain formation.

MSD data for the 7.5 ns simulation at 320K. The green dots are the data used for diffusion coefficients \( dx^2 + dy^2 + dz^2 \), the red circles are the \( dx^2 \), and the blue x’s are \( dy^2 \).

(a) Temperature vs. time for the run for comparison to temperature equilibration regime. (b) MSD with the best fit line shown. The excluded thermalization data are in red.

Arrhenius plot for a flat, smooth, three-layer gold slab. The parameters for the fit (with 95% confidence bounds in parentheses) are: slope = -0.142 (-0.1919, -0.092) eV, y-intercept = -7.462 (-9.304, -5.619).

Animation showing the general setup for the first site vacancy simulation. The sites where an atom is missing from the top layer are shown with the red circles. Note: The actual lattice is the hexagonal lattice, the schematic here is just for illustration of the general structural setup.
Three snapshots in time showing the stability of a molecule in a trench site. In all cases, the molecule that reacted with the trench is bordered with a red rectangle. (a) Shows the instant before the molecule becomes absorbed or “falls” into the trench. Note also that the molecules near the lower left trench approach it but do not appear to react to it, despite having several molecules near it. (b) Shows the molecule right after absorption into the trench. (c) Shows the system at a much later time (over 1 ns later, a relatively long time for simulations), and the molecule is completely stable in the trench. A long cluster seems to form off the anchored molecule, although one can see that it is not very organized and it was mobile.
ABSTRACT

Molecular Dynamics Simulations of a Pentacene Derivative on Gold Surfaces in the Quantum Mechanical and Classical Mechanics Frameworks

by

Ryan Miller

University of New Hampshire, May, 2016

With increasing potential to serve as a basis for constructing organic photovoltaic devices, the directed self-assembly of small organic molecules on metal substrates provides an interesting new method of bottom-up device building. By utilizing the qualities inherent in the substrate and molecules, one can manipulate surface topology to create functional layers of molecules in a desired configuration. However, tailoring these organic layers in a useful way requires detailed analysis of the substructure characteristics.

With top-down visualization approaches like Scanning Tunneling Microscopy (STM) providing a limited view of assembled layer structural characteristics, molecular dynamics simulations must be employed to obtain a more detailed depiction of how the structures form. With atom-scale spatial resolution and femtosecond timescale resolution, molecular simulation provides a means to study surface diffusion and self-assembly with a clear view of the molecular trajectories and atomic movements, and several virtual experiments can be conducted with easy manipulation of initial conditions. In this study, several arrangements of
the pentacene derivative 5,6,7-trithiapentacene-13-one (TTPO) were studied on flat Au(111) surfaces and Au(788) steps. The results obtained agree with those previously obtained in STM experiments for TTPO on these surfaces. Also, details of the diffusive properties of TTPO on gold were obtained from analyzing the trajectories at varied temperatures. Finally, different arrangements of surface defects gave insight into TTPO’s interaction with substrate vacancies. The study bridges theoretical simulation with past experiment and provides insights into phenomena previously unstudied for this molecule-substrate complex.
Chapter 1

Introduction and Motivation

1.1 Organic Photovoltaic Devices

Recent years have shown a promising upward trend in efficiencies of organic photovoltaic devices. The National Center for Photovoltaics (NCPV), a subgroup of the National Renewable Energy Laboratory (NREL), released data showing trends in solar cell device efficiencies, shown in Figure 1-1. With their potential to provide cheaper, more flexible, and more efficient devices to convert solar energy into electrical energy, organic photovoltaics have become an increasingly interesting area of study in solid-state physics.
As its name suggests, organic photovoltaics (OPVs) are a subclass of solar energy conversion cells whose semiconducting components are composed of organic molecules. Currently, the most efficient solar cells are composed from relatively expensive non-organic materials like silicon and gallium arsenide, as can be seen in Figure 1-1. While these materials have provided exciting breakthroughs in making efficient devices, high manufacturing costs hinder their attractiveness to be the candidates for the future of mass solar cell production, especially as the demand for solar energy conversion devices goes up. Therefore, organic photovoltaics provide a hopeful future to serve as the basis for devices that are able to be produced on a large scale at a lower cost than their inorganic counterparts.

Figure 1-2 shows a simplified schematic of the underlying idea of a bilayer OPV.
The underlying idea for an OPV is that a photon hits the surface of either the donor species, and is subsequently absorbed, creating an exciton. As shown in Figure 1-2, this exciton then diffuses to the donor/acceptor (D/A) interface, and the electron then jumps from the conduction band of the donor molecule to the conduction band of the acceptor species (lower energy than that of the donor). After this charge separation, the electron diffuses through the acceptor species to the cathode, and the positively charged hole diffuses to the anode, causing a current to flow. There are several factors here that can be tailored in order to achieve maximum efficiency from the system. For the bilayer scheme shown in Figure 1-1, the donor/acceptor layers must be chosen such that (i) They are not too narrow such that photon absorption is hindered and (ii) They are not too wide such that the exciton
cannot migrate to the D/A interface, and the photon is just re-emitted. The system must also be constructed in a way that is mindful of the energy differences between the donor and acceptor species, so that charge separation can occur at the interface. This project focuses on a promising donor species candidate (p-type semiconductor), 5-6,7-trithiapentacene-13-one (TTPO). Its assembly on a gold surface (possible anode surface) is studied with promising conclusions for OPV applications.

1.2 Pentacene and TTPO

1.2.1 Pentacene

Of the diverse sets of organic semiconductors, a very highly-studied candidate for an OPV donor species is pentacene [11, 12]. The name explains its structure: penta- “five,” and acene- “fused benzene rings.” It is a rigid, planar molecule with a molar mass of 278.3 g/mol, and a band gap of 2.2 eV [13]. While pure pentacene itself serves as an effective donor molecule, a new class of pentacene derivatives with substituents at the 6,13 positions have emerged that aim to tailor the base molecule towards more effective electrical and structural properties. In addition to these properties, a goal of a pentacene derivative is to decrease the photo-oxidation susceptibility. In order to manufacture a device in air (cheaper than having to be in a vacuum environment), it is important that the donor and acceptor species have a high resistance to O₂, which pentacene does not. This oxidation reduces charge-carrying ability, hindering performance of an OPV.
1.2.2 TTPO

TTPO is an interesting new molecule with promising potential to serve as a good nanoscale semiconductor, essentially a molecular wire for nanoscale electronic systems. It consists of a pentacene base (5 benzene rings in a line) with an oxygen atom double bonded to the center carbon atom, with three sulfur atoms bonded on the opposite side of the benzene backbone from the oxygen. The molecular structure of TTPO is shown in Figure 1-3 [14].

![Molecular structure of TTPO](image)

Figure 1-3: Molecular structure of TTPO

TTPO has an energy gap between its highest occupied molecular orbitals and lowest unoccupied molecular orbitals (HOMO-LUMO gap) of 1.90 eV [6], and possesses an inherent polarity. The dipole moment points from the (positive) central sulfur atom to the (negative) oxygen atom. The S-Au interaction strength has been characterized to be on the order of 1.95 eV [15]. The interaction of the sulfur bridge with the gold surface is complicated. Since the Au-Au bond dissociation energy is $2.34 \pm 0.0051$ eV [16] (very close to the S-Au “bond” strength), the sulfur bridge has the ability to potentially alter surface topology of the metal substrate. In addition, it will break the symmetry (about the long axis) associated with pentacene, such that TTPO will have a lateral tilt associated with its assembly. This strong attraction dictates the adsorption of the molecule on the metal surface and will, therefore, be a crucial aspect of self-assembly.
Thermal Stability

A problem with current dominant organic semiconducting molecules is their susceptibility to breakdown at high temperatures. This is an undesirable trait because it limits the operating temperatures of devices containing the organic material. For example, a common class of organic semiconducting molecules studied are acene and acene derivatives. An organic field effect transistor (OFET) was constructed using 6,13-bis(triisopropylsilylethynyl)pentacene (TIPS-pentacene), but only withstood temperatures up to 120°C before degradation occured [17]. Pure pentacene was shown to have an increased mobility with increased temperature, but this was again done at low-temperatures (up to 160°C) [18]. TTPO has a promising thermal stability, with a melting point in air of 386°C- 388°C, and is indefinitely stable in acid-free solution [4]. The ability to maintain its structural integrity at high temperatures makes TTPO an excellent candidate for use in high-temperature devices such as sensors and solar cells.

Crystallization and Solubility

In addition to being thermally stable at high temperatures, TTPO also readily crystallizes, making it a more convenient candidate to use in thin-film devices, and is soluble in many different solvents. It also crystalizes in a head-to-tail fashion (such that the sulfur bridge is above the oppositely-charged underlying oxygen), leading to $\pi - \pi$ orbital overlap and increased charge-carrying ability. The molecule’s solubility in chloroform, dichloromethane, THF, DMF, o-dichlorobenzene, and 1,2,4-trichlorobenzene enable it to be useful in applications that range from spin-coating to printing [4]. This ability to be in “ink” form makes TTPO more attractive to use in flexible electronics, a desirable trait for solar cells and other emerging nanotechnology applications.
Synthesis Procedure

The relative ease of synthesis of TTPO also makes it a more competitive molecule to use for organic semiconducting applications. TTPO has been synthesized in two different ways, both using 6,13-pentacenequinone as a precursor. From this precursor it is reduced to 6(13H)-pentacenone, and then dissolved in N,N-dimethylformamide or 1,2,4-trichlorobenzene and reacted with elemental sulfur to produce TTPO, or it is reduced to 6,13-dihydro-6,13-pentacenediol using NaBH$_4$, and then dissolved and reacted with sulfur in the same way to produce TTPO [4]. It is crucial that an organic semiconducting molecule is easy to synthesize so that manufacturing is cheaper and easier than its inorganic counterparts.

1.3 Molecular Dynamics Simulations

A common objective in surface chemistry, surface physics, materials science, biophysics, and other fields is to understand the structural and dynamical behavior of molecules deposited on a substrate. There is much to be learned from how molecules react and assemble on a surface. For example, a common employment of MD simulation for molecules on a surface in the biology realm is to study protein structure and folding [19, 20]. Performing MD is an impossible task on paper, as one must account for the interatomic forces between all atoms in the system, consider the ground state configuration of the electrons in the system, etc. Therefore, molecular simulation exists as a subfield of chemistry, physics, biology, material science, and more to obtain more detailed information about a system that is unable or difficult to obtain via top-down or other visualization techniques. In addition to visualization resolution at the atomic level, molecular dynamics simulations allow one to analyze systems at the pico- or nanosecond timescales, so that detailed analysis of timescales much shorter
than able to be processed in real time can be performed.

This project aims to perform detailed molecular dynamics analysis on TTPO on Au systems comparable to those studied via STM [6]. This approach to studying these systems have not been performed and, when coupled with the experimental evidence listed, will provide a more thorough and comprehensive characterization of this interesting system. This work is at the crossroads of physics, chemistry, material science, and computer science, and therefore contains background and analysis from an interdisciplinary list of sources. The scope of the project extends from one and two-molecule systems on a small gold slab to monolayer coverages of many-thousand atom unit cells for timescales up to several nanoseconds. This time frame and system size were determined to be sufficiently large as to sample an appropriate number of molecules, and not too large such that it becomes computationally impractical. Also, by using this scope, a combined Density Functional Theory (DFT) and classical mechanics study can be performed, since DFT is very limited in what size systems can be explored.
2.1 First-Principles Methods

2.1.1 The Electron Density

When modeling molecules, crystals, interfaces between the two, or any other atom-scale system, one needs a way to take into account the quantum mechanical effects of the electrons in these many-body systems. The time-independent Schrödinger Equation for an N electron system is

\[
H \Psi = \left[ -\sum_{i=1}^{N} \frac{\hbar^2}{2m} \nabla_i^2 + \sum_{i=1}^{N} V_{ion}(\vec{x}_i) + \sum_{i<j}^{N} U_{int}(\vec{x}_i, \vec{x}_j) \right] \Psi = E \Psi, \tag{2.1}
\]

where the first term on the right hand side of Equation 2.1 is the usual kinetic energy term, \(V_{ion}\) is the external (to the electron) potential provided by the nuclei of the atoms, and \(U_{int}\) is the interaction term from other electrons. The interaction term complicates the problem because, if it were not there, one could separate the problem into many one-electron Schrödinger equations.

To circumvent this issue, DFT employs as its key component the electron density function \(n(\vec{x}_i)\). It is a function of three spatial coordinates. For an N-electron system, the electron
density can be written as

\[ n(\vec{x}) = N \int \Psi^* (\vec{x}, \vec{x}_2, ..., \vec{x}_N) \Psi (\vec{x}, \vec{x}_2, ..., \vec{x}_N) d\vec{x}_2 d\vec{x}_3 ... d\vec{x}_N \tag{2.2} \]

which, for a normalized wavefunction \( \Psi \), gives the probability density of finding an electron in the volume \( d\vec{x} \).

### 2.1.2 The Hohenberg-Kohn Theorems

DFT rests on two pillars, developed by Pierre Hohenberg and Walter Kohn in 1964 [21]. The Hohenberg-Kohn theorems can be stated as follows:

**H-K Theorem 1:** The electron density \( n(\vec{x}) \) uniquely determines the external potential \( V_{\text{ion}}(\vec{x}) \) of the system, and thus the Hamiltonian operator \( \hat{H} \) and the total energy \( E \) of the system. Therefore, the many-particle ground state is uniquely determined by the electron density.

The total energy of the system can be written as a functional of the electron density

\[ E[n(\vec{x})] = \int n(\vec{x}) V_{\text{ion}} d\vec{x} + T[n(\vec{x})] + E_{ee}[n(\vec{x})] = \int n(\vec{x}) V_{\text{ion}} d\vec{x} + F[n(\vec{x})] \tag{2.3} \]

\( E[n(\vec{x})] \) is the total energy functional, and \( F[n(\vec{x})] \) is the sum of the kinetic energy and electron-electron interaction energy terms. The functional \( E[n(\vec{x})] \) is considered to be the holy grail of DFT, as knowing its exact form would allow the many-body Schrödinger equa-
tion (Equation 2.1) to be solved exactly. However, there are two important things to note about these functionals:

1. The exact form of $E[n(\vec{x})]$ for a general system is not known. The only case in which this functional is exactly known is the homogeneous non-interacting electron gas (as given by the Thomas-Fermi model [22, 23]). All other systems are based on an approximation of this system (discussed below).

2. The functional $F[n(\vec{x})]$ is universal, i.e. it is independent of the system at hand. A major drawback of DFT is that the exact form of this universal functional for an interacting system is unknown. However, various approximation schemes exist to minimize the energy functional, as discussed below.

H-K Theorem 2: The electron density $n(\vec{x})$ that results in the ground state energy of the system is the true density if and only if it delivers the functional $E[n(\vec{x})]$ that corresponds to the ground state energy. For the systems in this study, there will be many states that satisfy this criterion, so the density will correspond to a local minimum.

In other words, the trial density $n_t(\vec{x})$ that gives rise to the Hamiltonian $\hat{H}_t$ (and a state $|\Psi_t\rangle$), where $\hat{H}_t$ and $|\Psi_t\rangle$ satisfy

$$
\langle \Psi_t | \hat{H}_t | \Psi_t \rangle = T[n_t(\vec{x})] + E_{ee}[n_t(\vec{x})] + \int n_t(\vec{x})V_{\text{ion}}d\vec{x}
$$

will be the true density of the system if and only if
\[ \langle \Psi_t | \hat{H} | \Psi_t \rangle = \langle \Psi_0 | \hat{H} | \Psi_0 \rangle = E_0 [n_t(\vec{x})] \tag{2.5} \]

where \( E_0 \) and \( |\Psi_0\rangle \) are the ground state energy and ground state wavefunction, respectively.

### 2.1.3 The Kohn-Sham Equations

While the above formulation of DFT is accurate, it is complex to solve for real systems. There are complicated effects from the spin of the electrons that make the problem difficult to solve when coupled with the other parts. In 1965, Walter Kohn and Lu Jeu Sham reformulated the H-K Theorems in a non-interacting framework [24]. In this method, the electrons are considered as a non-interacting gas moving in an effective potential \( V_{eff} \). They separated the functional \( F[n(\vec{x})] \) into three parts:

\[ F[n(\vec{x})] = T_s[n(\vec{x})] + V_H[n(\vec{x})] + E_{XC}[n(\vec{x})] \tag{2.6} \]

In this way, the total energy is separated into a kinetic energy part and an effective potential part, so that it will obey the K-S equations:

\[ (-\frac{\hbar^2}{2m} \nabla_i^2 + V_{eff}(\vec{x}))\Psi_{KS,i} = E_i \Psi_{KS,i} \tag{2.7} \]

where \( V_{eff} \) is called the Kohn-Sham potential, and \( \Psi_{KS} \) are the Kohn-Sham orbitals. These orbitals satisfy the original density function

\[ n(\vec{x}) = \sum_i^N |\Psi_{KS,i}|^2 \tag{2.8} \]

By separating the functional in Equation 2.3, the spin effects of complex electron-electron
interactions are isolated to the exchange-correlation (XC) term (the third term in Equation 2.6), and the first two terms can be dealt with easily. The XC term can be approximated to good accuracy, and the approximation used here is discussed in the next section. The Kohn-Sham potential can be further subdivided into its constituent pieces

\[ V_{eff} = V_{ion} + V_H + V_{XC} \] (2.9)

Here, \( V_{ion} \) is again the external ionic potential, \( V_H \) is the Hartree potential, which is from the Coulombic electron-electron interactions, and can be expressed as

\[ V_H = e^2 \int \frac{n(\vec{x}')}{|\vec{x} - \vec{x}'|} d^3 \vec{x}' \] (2.10)

and \( V_{XC} \) is the exchange-correlation potential, which is the sum of the exchange potential and correlation potentials, both of which arise from the Pauli Exclusion Principle.

**Local Density Approximation**

The exchange-correlation potential is the element of DFT that requires approximation, as opposed to the exact solution that solving the coupled set of Schrödinger equations would yield. One of the most popular methods of approximating this potential is the Local Density Approximation (LDA). In the LDA, the \( V_{XC} \) term in the Kohn-Sham potential can be approximated using the exchange-correlation energy per particle of a homogeneous electron gas, which is the only system that has an exact known form of the exchange-correlation functional. Using the LDA, the exchange-correlation energy \( E_{XC} \) can be approximated as a functional of the electron density \( n(\vec{x}) \) as
\[ E_{XC}^{LDA} = \int \epsilon_{XC}(n(\vec{x}))n(\vec{x'})d^3\vec{x'} \]  

(2.11)

where \( \epsilon_{XC} \) is the (known) exchange-correlation energy of the homogeneous electron gas. So, the LDA isolates each point and sets the value of the XC potential at that point to the value of a homogeneous electron gas of that density. From \( E_{XC}^{LDA} \), one easily obtains the exchange-correlation potential

\[ V_{XC} = \frac{\partial E_{XC}^{LDA}(n(\vec{x}))}{\partial n(\vec{x})} \]  

(2.12)

Using this approximation, every term in \( V_{KS} \) in Equation 2.9 is now known or approximated, and one can solve the Kohn-Sham equations for the system.

### 2.1.4 The Self Consistent Field Loop

In order to solve the Kohn-Sham equations in Equation 2.7, the electron density \( n(\vec{x}) \) is needed. However, \( n(\vec{x}) \) is a function of the Kohn-Sham orbitals, which are obtained from solving the KS equations. The process is coupled, which mandates an iterative “guess and check” method to obtain a solution. This process is called the Self Consistent Field Loop (SCF) method, sometimes called the Hartree-Fock method. Figure 2-1 diagrams the SCF loop.
The process starts with $V_{\text{ion}}$ as an input parameter that is defined by the system. Then, a pseudo-random $n(\vec{x})$ is generated (pseudo because the electrons are usually relatively localized initially, i.e. they are not in a totally random orientation). Using this initial guess, the effective potential $V_{\text{eff}}$ is calculated, and then plugged into the KS equations to obtain a $\Psi_{KS}$. Using the probability definition that

$$n(\vec{x}) = \sum_i |\Psi_{KS,i}(\vec{x})|^2$$

(2.13)
a new $n(\vec{x})$ is calculated, and if $|n_{\text{initial}}(\vec{x}) - n_{\text{final}}(\vec{x})|$ is within a certain threshold (defined by the user), then the loop exits and the $n_{\text{final}}(\vec{x})$ is used to calculate forces and other parameters of interest.

### 2.1.5 Born-Oppenheimer Molecular Dynamics

A very useful ab initio method of molecular dynamics that exploits useful properties from DFT is Born-Oppenheimer Molecular Dynamics (BOMD). The distinguishing characteristic of this MD method is that the electronic and nuclear degrees of freedom are separated and treated individually. Due to the large difference in evolutionary time scales for nuclei and electrons [25], this approach decouples the two and evolves each separately.

Consider the Hamiltonian for the system containing some set of atomic nuclei (at positions $\vec{R}$) and surrounding electrons (positions $\vec{x}$)[26]

\[
H = \sum_i \frac{\hbar^2}{2m_e} \nabla_i^2 - \sum_i \frac{\hbar^2}{2M} \nabla_i^2 \sum_R \frac{Z_i e^2}{|\vec{x}_i - \vec{R}_I|} + \sum_{i<j} \frac{e^2}{|\vec{x}_i - \vec{x}_j|} + \sum_{I<J} \frac{e^2 Z_I Z_J}{|\vec{R}_I - \vec{R}_J|}
\]  

(2.14)

The dynamics of the coupled system will be described by the time-dependent Schrödinger Equation (TDSE):

\[
i\hbar \Psi(\vec{x}, \vec{R}, t) = H \Psi(\vec{x}, \vec{R}, t)
\]  

(2.15)

BOMD assumes that one can separate $\Psi(\vec{r}, \vec{R}, t)$ into a product of decoupled states that represent the nuclei and electrons independently.
\[ \Psi(\vec{x}, \vec{R}, t) = \psi(\vec{x}, t) \phi(\vec{R}, t) \]  

Equation 2.15 can now be solved with the separated-state form of \( \Psi(\vec{x}, \vec{R}, t) \). The idea is to fix the nuclear positions for a given time and allow the electrons to reach self-consistency, then evolve the nuclei in a way that corresponds to the force on the nuclei. They will evolve according to Newton’s second law

\[ M \ddot{R}_I = -\nabla_I \langle \Psi_0 | H(R_I) | \Psi_0 \rangle = F_I \]  

(2.17)

where \( \Psi_0 \) is the ground state total wavefunction for a set of fixed nuclear positions with corresponding ground state energy \( E_0 \), and \( F_I \) is the force on the nuclei. This can be rewritten using the Hellman-Feynman Theorem

\[ -\nabla_I \langle \Psi_0 | H | \Psi_0 \rangle = -\frac{\partial}{\partial R} \langle \Psi_0 | H | \Psi_0 \rangle = -\langle \Psi_0 | \frac{\partial H}{\partial R} | \Psi_0 \rangle \]  

(2.18)

Taking the partial derivative of \( H \) with respect to \( R \) and integrating over the ground state \( |\Psi_0\rangle \) gives

\[ -\langle \Psi_0 | \frac{\partial H}{\partial R} | \Psi_0 \rangle = -\sum_{R_J} \nabla_{R_I} \frac{Z_{R_I} Z_{R_J} e^2}{|R_I - R_J|} + \int n(\vec{x}) \nabla_{R_I} \frac{Z_{R_I} e^2}{|\vec{x} - \vec{R}_I|} d\vec{x} \]  

(2.19)

where the definition of the electron density \( n(\vec{x}) \) has been used. Given this force, the equations of motion (Equation 2.19) can be integrated and solved for a new set of ionic positions. This completes one iterative cycle of BOMD. Graphically, this adds an outer loop to the SCF structure in Figure 2-1, which is shown in Figure 2-2:
In the ionic relaxation process, self-consistency must again be reached first for fixed ionic positions. Once a self-consistent electron density is reached for the initial ionic configuration, the ions are moved in a way that lowers the total energy of the system, and the process repeats. This relaxation loop continues until a threshold on the total energy and total force on the atoms are met so that the system is at the lowest energy configuration (the ground state of the system). The system is now in at least a local minimum and one can proceed with calculating parameters of interest.

2.1.6 Car-Parrinello Molecular Dynamics

BOMD is a useful tool but it has major computational drawbacks. Since the electrons are relaxed to the ground state at each time step, each requiring an SCF cycle composed of many
iterations, the process is relatively inefficient. Performing a dynamics simulation that covers interesting timescales requires hundreds or thousands of time steps, which makes BOMD a less-desirable candidate for performing dynamics simulations. Developed in 1985 by Robert Car and Michele Parrinello, Car-Parrinello Molecular Dynamics (CPMD) is an attractive alternative to BOMD. In CPMD, the electrons are relaxed initially to their ground state, then evolved simultaneously with the ions while being kept close to the ground state. To accomplish this, CPMD employs a trick: assign to the orbitals a fictitious “mass” \( \mu \) (really an inertia parameter, it has units of energy times time squared). This gives the orbitals a fictitious kinetic energy, corresponding to a low temperature, which acts to accelerate the electrons towards their ground state at each timestep. This adiabatic process ensures that the electrons remain close to their ground state while being evolved in tandem with the ions. The process dramatically decreases CPU time for calculations, making it the go-to method for many dynamics simulations in surface physics, biology, and more [27, 28, 29, 30].

As mentioned above, the orbitals in CPMD are assigned a fictitious mass \( \mu \) so that they may be evolved along with the ionic degrees of freedom. This is justified because the orbitals remain at a low temperature (through a proper choice of \( \mu \)), so they evolve adiabatically with the ions without transferring energy and remaining close to their ground state. This is in contrast with BOMD, in which the electrons are self-consistently relaxed at each timestep. Therefore, initial relaxation to a proper minimum is crucial. The method relies on an extended Lagrangian that takes into account these orbital kinetic degrees of freedom. The Lagrangian can be written as
where $E[n(\vec{x})]$ is the Kohn-Sham energy density functional discussed earlier, $I$ indicates a sum over ions, and $i$ indicates a sum over electron orbitals, which are ensured to be orthogonal by the last term, which enforces the constraint

$$
\langle \psi_i(\vec{x}) | \psi_j(\vec{x}) \rangle = \delta_{ij}
$$

Finding the stationary points of the Lagrangian under variations of the parameters $\langle \psi \rangle$ and $R_I$ will yield the equations of motion for the ions and electrons. This is done via the usual Euler-Lagrange equations for $\langle \psi \rangle$ and $R_I$ [31]:

$$
\psi_i: \quad \frac{d}{dt} \frac{\partial L}{\partial \dot{\psi}_i^*} = \frac{\partial L}{\partial \psi_i^*} \Rightarrow \mu \ddot{\psi}_i = \frac{\partial E[n(\vec{x})]}{\partial \psi_i^*} + \sum_{i,j} \Lambda_{ij} \left( \int \psi_i^*(\vec{x}) \psi_i(\vec{x}) d\vec{x} - \delta_{ij} \right)
$$

where $\Lambda_{ij}$ is a matrix of Lagrange multipliers used to enforce the holonomic orthonormality constraint of $\psi_i$ above. This is necessary because orthonormality is not conserved in general by the Verlet algorithm. In practice, the system is evolved according to the equations of motion, and the orthonormality constraint is enforced at each timestep (orthonormality is enforced after the Verlet algorithm is applied).

$$
\vec{R}_I: \quad \frac{d}{dt} \frac{\partial L}{\partial \dot{\vec{R}}_I} = \frac{\partial L}{\partial \vec{R}_I} \Rightarrow M_I \ddot{\vec{R}}_I = -\nabla_I E[n(\vec{x})] = F_I
$$
This is the set of standard equations of motion for classical bodies that one would expect.

2.2 Classical MD

First-principles calculations, those obtained from quantum mechanical methods such as DFT (like those described above), are considered to be among the most accurate forms of molecular dynamics simulation. They consider electronic degrees of freedom, so that detailed information about the electron density of states is provided, giving a true picture of the ground state of a system. However, this process is inherently costly in regards to computational workload. For example, to simulate 10,000 timesteps on 40 nodes (each node containing 32 processor cores for a total of 1,280 processors) took 6 days, 17 hours, and 46 minutes of CPU time. Each timestep simulated 0.25 femtoseconds, so the entire calculation represented 2.5 picoseconds of real time. Also, the simulation only contained 252 atoms (3 layers of Au substrate and 1 TTPO molecule), a relatively small unit cell (see Chapter 3 for a detailed description of the computational specifications of the calculations in this project). For this study, diffusion of multiple particles across large surfaces over long timescales is pursued, justifying the need for an alternate method of simulation.

An alternative to modeling a system using first-principles interactions is to model the system classically. In classical MD simulations, pairwise interactions are computed directly through analytic equations, with user-specified parameters characterizing each atomic interaction. In this way, electronic degrees of freedom can be considered to be initially integrated out [26], and the problem of solving the N-body (for N electrons) Schrödinger equation and the trajectories of the nuclei collapses to solving classical Newtonian equations for the tra-
jectories of the atom as a whole. Atoms are represented as point particles that interact with other atoms via user-defined potential energy functions that best represent the actual interaction. For example, bonds and angular displacements can be modeled as springs, with the restoring “spring constant” and equilibrium length (or angle) being parameters set by the user. These parameters can come from experiment or from fitting through quantum mechanical simulations, and should be chosen carefully as to accurately reflect the physical nature of the interaction.

Classical molecular dynamics simulations can be performed with CPU times orders of magnitudes less than the full quantum mechanical calculations [32], and this improvement gets better with increasing system size. However, there naturally exists a trade-off. Using these classical potential methods introduces an inevitable degree of approximation. The justification to use classical simulations depends on the interests of the project. Here, its use is justified because, as mentioned above, long-term trajectories of several molecules (several dozen or more) over long time scales (hundreds of picoseconds or even nanoseconds) for many different temperatures are to be studied. Since computational effort for quantum mechanical calculations can increase to the third power with atom size [33] or even more, adding even one TTPO molecule or 1 layer of Au substrate can significantly increase the CPU time. Also, since classical parameters can be carefully be chosen to accurately reflect the physical behavior of a system, using classical dynamics will not significantly affect the quantities of interest in this study (diffusion coefficients, anchoring ability from site and edge defects, etc.). Therefore, the primary method of calculation shall be classical MD, with full quantum mechanical calculations used mostly to verify the accuracy of the parameterization and for comparison of small phenomena. Details of specific pairwise interactions and bonded/ non-bonded interactions to construct the force field in this study are summarized in Chapter
2.3 Integrating the Equations of Motion

After the forces acting on each atom (or ion) is calculated, there still remains the task of advancing the trajectory, or integrating the equations of motion. Whether it is a first-principles calculation or classical-mechanics based calculation, the force on an ion (or atom) will be of the form given by Newton’s Second Law:

$$F_I = -\nabla V(\vec{x}_I(t)) = M\ddot{\vec{x}}_I(t)$$  \hspace{1cm} (2.21)

In an MD simulation, one would like to know the trajectory of the particle for each timestep. So, Equation 2.21 needs to be integrated to find $\vec{x}_I(t)$.

2.3.1 The Verlet Algorithm

A common and useful method of integrating $\ddot{\vec{x}}(t)$ to find $\vec{x}(t)$ is to use the Verlet algorithm. This is a standard method of integration that various MD software use to evolve the atoms in an MD simulation. This method is standard because it has relatively low total error and only requires the two previous positions to calculate the next in the series. The algorithm starts by Taylor expanding the position at a timestep ahead of time $t$, and a timestep before time $t$

$$\vec{x}(t + \Delta t) = \vec{x}(t) + (\Delta t)\dot{\vec{x}} + \frac{1}{2}(\Delta t)^2\vec{x} + \frac{1}{6}(\Delta t)^3\vec{x} + \mathcal{O}(\Delta t^4)$$  \hspace{1cm} (2.22)
\[ \mathbf{x}(t - \Delta t) = \mathbf{x}(t) - (\Delta t)\mathbf{\dot{x}} + \frac{1}{2}(\Delta t)^2\mathbf{\ddot{x}} - \frac{1}{6}(\Delta t)^3\mathbf{\dddot{x}} + \mathcal{O}(\Delta t^4) \]  

(2.23)

Adding these two equations gives

\[ \mathbf{x}(t + \Delta t) = 2\mathbf{x}(t) + (\Delta t)^2\mathbf{\ddot{x}} - \mathbf{x}(t - \Delta t) \]  

(2.24)

So, given an initial position \( \mathbf{x}(0) \), the trajectory of every atom can be calculated, and the path of the molecule over time can be traced out. In a more traditional manner that illustrates its iterative nature, the algorithm looks like

\[ \mathbf{x}_{n+1} = 2\mathbf{x}_n + (\Delta t)^2\mathbf{\ddot{x}}_n - \mathbf{x}_{n-1} \]  

(2.25)

Starting at \( n=1 \) and an initial condition \( \mathbf{x}_0 \) for each atom, the algorithm can be solved for every position at any time for all the atoms. The Verlet method for solving the equations of motion is accurate up to fourth order in \( \Delta t \), which is very small (\( \Delta t \) is on the order of femtoseconds typically).

2.4 Thermostat Background

2.4.1 The Langevin Equation and Brownian Motion

When heating a system or keeping the temperature fixed, thermal energy is placed into the available degrees of freedom for the atoms involved. The Equipartition Theorem states that the energy is shared equally amongst the various degrees of freedom. For an ideal (non-interacting) monatomic gas, this is a simple notion: \( \frac{1}{2}k_B T \) goes into each translational
degree of freedom, so that the total thermal energy of the gas is $\frac{3}{2} N k_B T$. However, for a system of TTPO molecules (with each molecule containing complex degrees of freedom such as torsional potential energy, harmonic bonding energy, etc.), on a lattice of gold atoms (each with three translational degrees of freedom), this becomes more complicated.

To invoke a structured and quantitative description of diffusion properties at finite temperature via computer simulation, care must be used in deciding how to thermostat the system. Several methods exist to describe thermal contributions to the equations of motion, including the Nosé-Hoover Thermostat and Andersen Thermostat [34], both of which sample the NVT ensemble (coupling to a heat bath). There is no perfectly correct way to simulate a system, one only chooses a thermostat based on the qualities it provides and the quantities of interest in the study.

One way to thermostat a system is to treat the particles as placed in a fictitious solvent with a set friction parameter. The damping parameter is set to accurately simulate the desired temperature. For this study, this scheme (Langevin Dynamics) is used exclusively within the classical dynamics framework. In general, the Langevin equation for a free (no external potential) particle is given by [35]

$$m \frac{\partial \vec{v}(t)}{\partial t} = -\gamma \vec{v}(t) + \sqrt{2k_B T \gamma} \vec{F}(t)$$  \hspace{1cm} (2.26)

where the first term on the right hand side of Equation 2.26 is a viscous drag term (proportional to the velocity and a damping term $\gamma$), and the second is a term due to random collisions with the fictitious solvent particles and is proportional to the damping parameter and the temperature $T$. Since this term represents random motion, its time average will be equal to zero:
\( F(t) = 0 \) \hspace{1cm} (2.27)

Since the particles in this study interact with the other atoms of the system through various potentials, the equation gets modified to give the full equations of motion:

\[
m \frac{\partial \vec{v}(t)}{\partial t} = -\frac{\partial V}{\partial \vec{x}} - \gamma \vec{v}(t) + \sqrt{2k_B T \gamma} \vec{F}(t)
\] \hspace{1cm} (2.28)

where \( V \) is the potential felt by the atom. The damping parameter \( \gamma \) is inversely related to a characteristic damping time of the system; A longer damping time means a slower rate of energy exchange with the fictitious solvent. So, a small choice of damping time leads to a highly damped system, i.e. the objects placed in a viscous solvent. This parameter \( \gamma \) is also directly related to the diffusion coefficient of particles, which is discussed in further detail in Chapter 4.

### 2.4.2 Nosé-Hoover Thermostat

Another scheme to control the temperature in the canonical ensemble in an MD simulation is by means of a Nosé-Hoover thermostat. As opposed to the Langevin formulation discussed above, the Nosé-Hoover (NH) thermostat leads to trajectories that are deterministic, as opposed to the above stochastic scheme (which requires a random number seed in the input, so two identical trajectories will not be produced from identical initial conditions). The following discussion of the thermostat is based on the derivation in [34]. The underlying idea of a Nosé-Hoover thermostat is the introduction of the fictitious variable \( s \), velocity \( \dot{s} \), and “mass” of an extended system \( Q \). In the Nosé (not NH yet) [36] formulation, the factor \( s \) acts to scale the timestep of the extended system such that (the tilde is used to denote
extended coordinate):

$$\tilde{dt} = sdt$$  \hspace{1cm} (2.29)$$

while the atomic coordinates are the same in both the original system and extended system. This new, extended Lagrangian is

$$\mathcal{L} = \sum_i^N \frac{m_i}{2} s^2 \dot{x}_i^2 - V(\bar{x}) + \frac{Q}{2} \dot{s}^2 - 3Nk_bT \ln(s)$$  \hspace{1cm} (2.30)$$

where Q is the artificial “mass” of the introduced system. This extended Lagrangian can be solved to give expressions for the real momentum and momentum of the introduced heat reservoir

$$p = \frac{\partial \mathcal{L}}{\partial \dot{x}} = ms^2 \dot{x}$$  \hspace{1cm} (2.31)$$

$$\tilde{p} = \frac{\partial \mathcal{L}}{\partial \dot{s}} = Q\dot{s}$$  \hspace{1cm} (2.32)$$

The real and extended coordinates are related by the relations

$$r = \tilde{r}$$

$$p = \tilde{p}/s$$

$$s = \tilde{s}$$

in addition to the time relation in Equation 2.29. Equations of motion can then immediately be calculated for the purely Nosé scheme for the real variables:
While the above formulation is correct, it is not useful for practical implementation in molecular dynamics simulations. In an MD simulation, the timestep is fixed and constant, so that the entire simulation run is broken up into equal intervals of time \( t_{\text{simulation}}/N_{\text{timesteps}} \). However, with the above formulation, the real time is skewed by the factor \( s \), so time-averaging to obtain certain kinetic properties of a simulation like diffusion coefficients will be over intervals of time that are variable. To avoid this, Hoover [37] reformulated the Nosé scheme to obtain the more useful (for MD simulations at least) NH thermostat. Hoover noticed that the fictitious variables only arose in the real equations of motion as the parameter

\[ \xi = \frac{s\bar{p}}{Q} \]

where \( \xi \) can be thought of as a friction parameter. Rewriting the equations of motion in terms of this new parameter gives the new equations of motion (Equation 2.33 is unchanged in this new formulation)

\[
\frac{d\bar{p}_i}{dt} = \overline{F_i} = -\frac{dV(\bar{x})}{d\bar{x}} - \xi \bar{p}_i
\] (2.35)

\[
\dot{\xi} = \left( \sum_i^N \frac{p_i^2}{m_i} - 3Nk_B T \right)/Q
\] (2.36)
There are a couple points of interest in Equations 2.35 and 2.36. First, the constant $\xi$ acts as a scaling factor for the term proportional to the atom’s velocity in the force equation, hence why it is regarded as a frictional term. Also, the quantity $\xi$ stops changing when the term in the parentheses in Equation 2.36 is 0, or when the kinetic energy is $3Nk_bT/2$. So, the coupling thermostat effectively adds a frictional force to the atoms to constantly tend their velocities towards the desired thermostat temperature $T$. In practice, the system’s temperature will fluctuate around the desired temperature, with certain values of the $\xi$ leading to more suppressed values of the oscillations than others.
3.1 Molecular Dynamics in Quantum ESPRESSO

The software suite used for the quantum-based calculations in this project is Quantum ESPRESSO (QE), which stands for opEn-Source Package for Research in Electronic Structure, Simulation, and Optimization, versions 5.1.1 and 5.1.2. QE is an open-source, ab initio (from first principles) electronic structure suite that is immensely useful for molecular structure and dynamics simulations. The package is particularly useful because of its utilization of DFT principles and pseudopotentials. Also, calculations for periodic structures, such as the ones studied in this project, are easily implemented with QE’s utilization of plane-wave basis sets.

3.1.1 Executables

QE has many executables that allow one to perform several types of nanoscale system calculations. The two that were used in this project were Plane-Wave Self-Consistent Field
(pw.x) and Car-Parrinello (cp.x). BOMD is run via specifying “calculation=’md’” in pw.x, and CPMD is run by specifying “calculation=cp” in cp.x.

**PWscf (pw.x)**

PWscf (Plane-Wave Self-Consistent Field) is a diverse set of programs within the QE distributed. While it has several executables that serve various purposes, such as transition state pathways, band structure, charge density mappings, and more, it was used in this project to run a self-consistent calculation, a Born-Oppenheimer MD simulation, or a structural relaxation calculation. Specifically, the executable to run is pw.x in either the batch submission script or command line. With the ‘scf’ calculation type chosen, it checks for initial self-consistency, meaning that the system is able to achieve a converged electron density. This can be a useful tool for providing a quick check on if the electron density of a certain initial geometry is able to achieve convergence. In addition, pw.x can optimize the geometry of the system to specified energy and force thresholds with the ‘relax’ calculation option. Figure 3-1 shows an example of an input file for an SCF calculation.
Figure 3-1: Example input file for an SCF calculation. Most atomic coordinates have been suppressed for brevity.

Input Parameters for PWscf

The input file for a PWscf calculation is separated into sections, also known as cards. A card begins with a ‘&’ and ends with a ‘/’. The ‘&’ precedes the namelist for that card, which is the object that is being described in that section. The items within each card
are called variables, and not all variables must be filled out for a calculation. There are three mandatory cards that must be at least partially filled out for a calculation in PWscf: &Control, &System, and &Electrons.

The variables for &Control set the type of calculation to be performed (calculation) along with details of where to store output files (outdir), and where to find pseudopotentials (pseudo). In addition, a dipole correction is created here by setting the variables “tefield” and “dipfield” to “true.” This creates a sawtooth external electric field through the system.

&System defines the system to be analyzed: Bravais lattice type (ibrav – the number corresponds to a specific Bravais lattice type within QE– orthorhombic is used here), unit cell dimensions (A x B x C), number of atoms in unit cell (nat), kinetic energy cutoff (ecutwfc/ ecutrho), and Fermi-level smearing (smearing). Smearing is needed in order to integrate the sharp discontinuity of electron density at the Fermi surface. In addition, the parameters of the dipole correction are set here: the direction of the external electric field (edir), the point of the discontinuity in the sawtooth field as a fraction of the length of the dimension of the unit cell set by edir (emaxpos) (0 < emaxpos < 1) (set to the middle of the vacuum region), the point in the unit cell where the sawtooth potential decreases as a fraction of the length of the dimension of the unit cell set by edir (eopreg) (0 < eopreg < 1), and the amplitude of the E-field (eamp).

&Electrons sets the convergence threshold for an SCF calculation (conv_thr), along with setting initial conditions for generating electronic wavefunctions (startingwfc). It also is where one sets the “mixing mode,” the amount of $n(\vec{x})$ from the previous iteration that is carried over to the next iteration. The default is 0.7, and lower amounts (0.3 or lower) can be used to help with convergence issues or to speed up calculation time.
**Additional variables that need to be set for relax calculation:** Table 3.1 shows the minimum extra variables that need to be set for a geometry relaxation calculation in the pw.x executable.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;Control</td>
<td></td>
</tr>
<tr>
<td>calculation='relax'</td>
<td>Sets the type of calculation to be performed</td>
</tr>
<tr>
<td>&amp;Ions</td>
<td></td>
</tr>
<tr>
<td>ion_dynamics='bfgs'</td>
<td>Uses Broyden-Fletcher-Goldfarb-Shanno algorithm to evolve ions, can use other algorithms such as verlet, damped and Langevin dynamics, and more</td>
</tr>
<tr>
<td>etot_conv.thr = 1.0D-4</td>
<td>Energy threshold for convergence</td>
</tr>
<tr>
<td>forc_conv.thr = 1.0D-3</td>
<td>Force threshold for convergence</td>
</tr>
<tr>
<td>ATOMIC_POSITIONS angstrom</td>
<td>Atomic coordinate information</td>
</tr>
<tr>
<td>Au 0 0 0 0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>Au 4.08 0 0 0 0 0</td>
<td></td>
</tr>
<tr>
<td>..................</td>
<td></td>
</tr>
<tr>
<td>Au 17.3 17.3 9.9 0 0 1</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** The list of variables discussed (for both SCF and relax calculations) is not exhaustive; Only the variables related to the work here are mentioned. A complete list of pw.x variables with descriptions can be found in [38].

In Table 3.1, most atomic coordinates have been suppressed for brevity, only a few are shown. The first entry is the atomic type, i.e. element name. The following three slots are for the x, y, and z coordinates, respectively. The final three slots are the constant to multiply the x, y, and z component of the force by, respectively. This can be either a 0 (force is set to zero in that direction, the atom is held stationary in that direction) or a 1 (atom is allowed
to move in that direction). For surface calculations, the bottom layer or few layers are held fixed to simulate coupling to the bulk solid, and to avoid complete reconfiguration of the surface.

cp.x

There is an executable within QE to perform CPMD, cp.x. Cp.x works in the same way as pw.x with three main differences:

1: There are new variables, nr1b, nr2b, and nr3b that must be set in the &System card for USPP. They represent the size of the augmentation charge, which is a way of exploiting the locality of charges to increase computational efficiency. The values were each set to 20.

2: The flats ndr and ndw must be set in the &Control card. They are numbers (usually 50 and above for convention) that indicate a certain run so that future runs can easily be restarted from intermediate steps. For example, the first run may have ndr = 50 and ndw = 51, so it would read the “from_scratch” restart option and write the output to a file “prefix.51”, for whatever prefix one has chosen for the simulation. The next simulation can have the flags ndr = 51, and ndw = 52 set so that it reads the output from the previous run and writes a new output to the new flag 52. This makes it easy to restart runs without having to start from scratch.

3: By default, the cp.x code only samples at the gamma point (k=0), so the K_POINTS card is no longer specified.
3.1.2 Periodicity

The Plane Wave Basis

Electrons in a repeating structure will obey a periodic potential (due to the ions)

\[ V(\vec{x}) = V(\vec{x} + \vec{L}) \]  \hspace{1cm} (3.1)

where \( \vec{L} \) is the length of the unit cell in direct space. A periodically-repeating structure also has a representation in reciprocal space (often called momentum space, k-space, etc.). It is often more useful to work in this space to calculate band structures, momenta, and energies. Bloch’s Theorem says that the energy eigenstates of a periodic system can be represented by Bloch waves, which have the wavefunction form

\[ \psi(\vec{x}) = e^{i\vec{k} \cdot \vec{x}} u(\vec{x}) \]  \hspace{1cm} (3.2)

for a periodically-repeating function \( u(\vec{x}) \) (which has the same periodicity as the crystal). \( u(\vec{x}) \) can be expanded into an infinite sum of plane waves

\[ u(\vec{x}) = \sum_{\vec{G}} C_{\vec{G}} e^{i\vec{G} \cdot \vec{x}} \]  \hspace{1cm} (3.3)

where the sum runs over all reciprocal lattice vectors \( \vec{G} \), and \( \vec{G} \cdot \vec{L} = 2\pi m \), for an integer \( m \). Inserting this into the equation for \( \psi(\vec{x}) \) gives

\[ \psi(\vec{x}) = \sum_{\vec{G}} C_{\vec{k} + \vec{G}} e^{i(\vec{k} + \vec{G}) \cdot \vec{x}} \]  \hspace{1cm} (3.4)

This is a sum over an infinite number of reciprocal lattice vectors, which is inefficient com-
putationally. To resolve this, QE only considers a finite number of user-specified points in the first Brillouin Zone (BZ) in calculations.

**The Unit Cell**

Paramount to accurately and efficiently representing a molecular system is the careful construction of its unit cell to be ran by QE. The unit cell must be chosen such that it contains enough information about the system that running it will give an accurate description of properties, but not contain too much detail that it overloads the memory capabilities of the machine executing the task. For each task with a unique initial configuration of atoms to be executed, a unique unit cell must be constructed. The unit cell must be constructed as to also preserve the periodicity of the crystal. It is important to note that the unit cell (supercell) must be large enough to not only contain and isolate the non-periodic structure, but also large enough such that the non-periodic structures do not interact with those in adjacent cells.

For the simulations performed in this project, the unit cell always contained a gold substrate with one or more TTPO molecules on it. The ideal scenario would be to place the molecule(s) on many layers of gold, since this would better represent the real systems studied by STM and other methods. However, adding even one layer of gold adds many atoms to the unit cell, which becomes computationally costly very quickly. So the balance must be chosen carefully: provide enough layers of gold to return an accurate calculation for your quantity of interest (the amount of layers depends on the quantity one is interested in), but do not provide too many such that it overloads the performance capabilities of the computing machine.

**Adding a dipole correction:** For surface calculations, the slab supercell can either be
symmetric (atoms are in the middle of the supercell with vacuum on both sides), or anti-
symmetric (surface on one side of the supercell and vacuum filling the space above or below 
it). Recall that the potential in a crystal is periodic

\[ V(\vec{x}) = V(\vec{x} + \vec{L}) \]

where \( \vec{L} \) is a direct lattice vector. However, an asymmetric supercell will have a different 
potential on one side of the supercell and the opposite side [39]. To fix this, a dipole 
correction must be added in the calculation. This sends a sawtooth potential through the 
unit cell (with the discontinuity falling in the vacuum region) that will ensure the periodic 
boundary condition is upheld. Table 3.2 shows the extra lines that must be added to a QE 
input file to create a dipole correction.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;Control</td>
<td></td>
</tr>
<tr>
<td>tefield = .true.</td>
<td>Creates a sawtooth potential to uphold boundary conditions</td>
</tr>
<tr>
<td>dipfield = .true.</td>
<td>Produces a dipole correction to ionic potential</td>
</tr>
<tr>
<td>/</td>
<td></td>
</tr>
<tr>
<td>&amp;System</td>
<td></td>
</tr>
<tr>
<td>edir = 3</td>
<td>Sets direction to create apply potential</td>
</tr>
<tr>
<td>(3 is z direction)</td>
<td></td>
</tr>
<tr>
<td>emaxpos = 0.5</td>
<td>Position in unit cell where discontinuity in sawtooth potential is to exist. (Must fall in middle of vacuum region and 0 &lt; emaxpos &lt; 1)</td>
</tr>
<tr>
<td>eopreg=0.06</td>
<td>Area in unit cell where sawtooth decreases</td>
</tr>
<tr>
<td>eamp = 0</td>
<td>Amplitude of external electric field</td>
</tr>
<tr>
<td>(0 for isolated slab)</td>
<td></td>
</tr>
</tbody>
</table>

It is important to note that the variables emaxpos and eopreg are unitless (they are a fraction of the unit cell), and both take on values restricted to between 0 and 1. The variable
eamp sets the value of the external electric field induced by the sawtooth potential (which changes during the self-consistency run to uphold the value of eamp). For an isolated slab, eamp=0.

3.1.3 K-Point Sampling and Ultrasoft Pseudopotentials

In order to get a convergent calculation, QE samples the system at user-defined k-points to calculate the total energy. Since things that are large in direct space are small in reciprocal space and vice versa, for sufficiently large systems it is enough to sample at only the gamma point.

As discussed above, each wavefunction can be expanded in an infinite number of plane waves. This is computationally costly, and so one must truncate the expansion at an energy $E_{\text{cut}}$, where

$$
\frac{\hbar^2|\mathbf{k} + \mathbf{G}|^2}{2m_e} < E_{\text{cut}}
$$

(3.5)

Figure 3-2 shows a two-dimensional BZ with $E_{\text{cut}}$ represented by the circle. All points within the circle are included in the plane wave expansion, and all points outside it are neglected. This is validated due to the fact that the expansion coefficients $C_{\mathbf{k+k'}}$ are large for small kinetic energies $\frac{\hbar^2|\mathbf{k} + \mathbf{G}|^2}{2m_e}$ and so expansion coefficients for larger kinetic energies do not contribute much to the expansion.
So, the number of reciprocal lattice vectors to sum over is limited by the user-specified cutoff kinetic energy.

Here, the cutoff kinetic energy is 30 Ry, and the cutoff kinetic energy for the charge density is chosen to be much larger at 400 Ry. These choices are were chosen to be satisfactory for the choice of pseudopotential used, which is a Vanderbilt Ultrasoft Pseudopotentials (USPP). Pseudopotentials are a way of modelling the complex effects of core electrons of atoms in plane-wave based simulations. As its name suggests, USPP is “soft,” meaning that it is a smoother function in the interior of the atom. Pseudopotentials match the actual potential exactly outside some cutoff radius (close to valence shell radius of atom).
3.1.4 Temperature Control

QE employs several methods of temperature control in pw.x, with most being different applications of a velocity-correction scheme. Thermostat settings are defined using the ion_dynamics variable in the &IONS card. Available options include different settings for direction velocity rescaling, Berendsen “soft” velocity rescaling, the Andersen thermostat, or no rescaling at all (uncontrolled temperature).

Experiments performed in the lab are typically represented by the canonical (NVT) ensemble. In lab conditions, it is the temperature that is often kept constant, not the energy. Recall the continuous partition function for the canonical ensemble

\[
Z = \frac{1}{h^3} \int e^{-\frac{H(\vec{x},\vec{p})}{k_B T}} d^3\vec{x} d^3\vec{p}
\]  

(3.6)

for the canonical positions and momenta \(\vec{x}\) and \(\vec{p}\). Since the system is allowed to be in thermal contact with the surroundings (usually ultra-high vacuum), each microstate will have a different energy, but will be at the same temperature \(T\). For this reason, the velocity-correction schemes in QE aim to control the temperature of the ions, rather than their energy.

In practice, the PWscf package allows one to adjust the velocities of the ions in the system to control the temperature. The simplest method of controlling temperature is with the ion_dynamics='rescale-v' method. This requires two other variables to be set in the &IONS card as well: tempw and nraise. The initial (and target) temperature is set by tempw, and every nraise iterations the velocities are adjusted to reset the temperature to tempw. This is the easiest way to keep temperature fixed. A similar method, ion_dynamics='rescale-T', multiplies the instantaneous temperature at the end of a step by the variable delta_t.
‘reduce-T’ will add the value -delta_T to the instantaneous temperature every nraise steps. The method to be used is dependent on how one desires to control temperature for the particular simulation, but there is a lot of freedom here to set the thermostat for dynamics.

### 3.2 LAMMPS

LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) is a powerful open-source molecular mechanics software developed, maintained, and distributed by Sandia National Laboratories [40]. It was chosen to be the classical dynamics simulation software for this project due to its extensive use and respect in the computational sciences, its well-documented user manual [41], and its wide range of potentials. It runs on machines that compile C++, such as the primary machine used for computation in this project, Trillian (see below), and has the Message-Passing Interface (MPI) library. LAMMPS was installed on Trillian and built with the optional packages USER-REAXC and RIGID, along with the standard packages. It was compiled using the Portland Group, Inc. (PGI) compiler. LAMMPS has the ability to simulate several thousands (even hundreds of thousands or more) of atoms for nanosecond timescales, so it is well-suited to simulate gold surfaces and diffusion of TTPO molecules in this project.

The following will be a brief introduction to the potentials used to model the systems in this project, and these descriptions with even more details can be found in [41]. For a given variable, there are many options called styles in LAMMPS. The parameters that characterize each style are called arguments for that style. For example, a typical set of lines that defines a pair interaction in LAMMPS is
Table 3.3: Example of Variable Input in LAMMPS

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
<th>Arguments</th>
</tr>
</thead>
<tbody>
<tr>
<td>pair_style</td>
<td>lj/cut 10.0</td>
<td></td>
</tr>
<tr>
<td>pair_coeff</td>
<td>1 2 1.0 4.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.3 shows a particular set of lines that define a 12/6 Lennard Jones interaction style with a cutoff radius (argument) of 10.0 in the first line (unit style is specified in input file header). The second line states that the interaction is between atom type 1 and 2, and the interaction has an $\epsilon$ value of 1.0 units, and a $\sigma$ value of 4.0 units. The same scheme is followed for bond styles and bond coefficients, dihedrals, impropers, atom styles, etc. More than one style for a given variable can be defined with the pair_style hybrid command. Note that the variables, values, and arguments are listed on the same line, separated by an arbitrary amount of spaces, and contain no units (only numerical or name input).

To perform a LAMMPS calculation, three files are involved: the input file, the data file, and the submission script file. The batch submission file is the usual file that points to the executable to run the program, says which file is the input file, defines the number of nodes, processes, etc. The input file contains information on pair interactions types (defines pair styles), bond types, boundary conditions (periodic, fixed, shrink wrapped), units, and simulation details. For example, the input file says which parameters (thermodynamic, trajectory, velocity, etc.) to dump to the output file, which is also defined here, how many timesteps to run, etc. The data file contains all topology information for the system, including atom positions, bonds, angles, dihedrals, impropers, and unit cell size. The pair and bond coefficients can be set in either the input file or the data file. See Appendix B for an example input file in LAMMPS with annotations.
3.2.1 Harmonic Bonds

Arguably the most popular bond style in LAMMPS is the harmonic bond. This treats bonded atoms in the usual classical paradigm of being connected by a spring. It is set with the bond_style harmonic command. Its exact functional form is

\[ E_{\text{bond}} = K_{\text{bond}}(r - r_0)^2 \]  

(3.7)

where \( K_{\text{bond}} \) is the effective spring constant of the harmonic bond (with the factor of 1/2 absorbed), and \( r_0 \) is the equilibrium length of the bond. Data for \( K_{\text{bond}} \) and \( r_0 \) can be fitted from QM calculations for the system at hand, or can be taken/derived from literature for each bond type. For example, the National Institute of Standards and Technology (NIST) [42] contains a database of characteristic infrared vibrational frequencies between pairs of atoms in different chemical environments, which can be used to calculate the usual spring constant (without the factor of 2) via the classical equation

\[ \omega = \sqrt{\frac{k_{\text{bond}}}{\mu}} \]

where \( 1/2k_{\text{bond}}=K_{\text{bond}} \), and \( \mu \) is the reduced mass of the interaction. For this project, however, parameters for harmonic bonds, angles, LJ potentials, and dihedrals were taken from the AMBER-FF94 force field [43], which is distributed free with the TINKER software package [44], and the general AMBER force field (GAFF), which is distributed with the AmberTools15 package [45]. A summary of the parameters for the various interaction types are summarized in Table 3.4. The parameters were chosen with care to most accurately reflect the chemical environment in which it exists. For example, a carbon-hydrogen bond will have
a different $K_B$ and $r_0$ if it is in an alkyl group versus an aromatic compound like benzene.

AMBER contains many different hybridizations and chemical setups to choose from. As TTPO contains a pentacene backbone (5 benzene-like rings), the aromatic parameters were chosen to good accuracy.

### 3.2.2 Angular Potential Energy

Similar to the spring-like potential energy stored in a chemical bond, there also exists a lowest-energy angle between three atoms joined by two bonds. Any deviation from this angle will result in an increase in angular potential energy (due to repulsion between electron clouds), and there naturally exists a potential energy equation that models this behavior. LAMMPS contains several styles to implement this energy, and the one chosen for this project is the harmonic potential, represented by the equation

$$E_{\text{angle}} = K_{\text{angle}}(\theta - \theta_0)^2$$

(3.8)

where $K_{\text{angle}}$ is the restoring energy constant associated with the particular angle. For TTPO, all angles between sets of three bonded atoms are $120^\circ$.

### 3.2.3 Dihedrals (Torsional Energy)

A set of four atoms connected by three bonds, all in the same plane, will have with it a potential energy that results by "twisting" an end atom out of the plane, as shown in Figure 3-3.
Figure 3-3: Figure showing a molecular dihedral. Displacing an end molecule into or out of the page results in a torsion about the bond between the middle two atoms, and a resulting harmonic potential energy. From [2].

This torsional energy is represented in LAMMPS by defining harmonic dihedrals that obey the energy equation

\[ E_D = K_D[1 + d\cos(n\phi)] \tag{3.9} \]

where \( K_D \) is the energy constant associated with the particular dihedral, \( d \) is \( \pm 1 \) (chosen in this project to be -1 to reflect a restoring energy), and \( n \) is an integer (chosen to be 2 here). There were 5 types of dihedrals defined for the TTPO molecule, which are listed and have the values for each summarized in Table 3.4.

### 3.2.4 Potentials (Pair Styles)

There is a vast array of potentials that exist to describe the interaction of pairs of atoms, both within LAMMPS and in other contexts. Some provide a higher degree of accuracy (and, therefore, usually computational cost), while others can be more simplistic and still provide a sufficient amount of detail. It is entirely up to the LAMMPS user to decide which potential(s) to use to describe the system at hand, as different potentials are more appropriate
for different chemical contexts. The following will provide a brief description of the specific potentials (pair styles) used to model TTPO on gold surfaces, and why each potential was chosen.

**Morse Potential:** The interaction between gold surface atoms and sulfur atoms in SAM structures is a debated issue in molecular mechanics simulations. Although various potentials exist to describe the Au-S potential in LAMMPS, the Morse potential has been shown to be a very successful potential in MD simulation to describe the sulfur-gold interaction [46, 47, 48, 49]. The Morse potential is given by the equation

\[
E_{\text{morse}} = D_0\left[ e^{-2\alpha(x-x_0)} - 2e^{-\alpha(x-x_0)} \right]
\]

(3.10)

where \(D_0\) is the characteristic bond energy, \(\alpha\) is a parameter with dimensions of inverse length (frequency), and \(x_0\) is the equilibrium bond length. The Morse potential is often preferred in modeling interactions like Au-S because it can simulate a partially covalent bond. While LAMMPS cannot simulate bond breaking and formation explicitly, the Morse potential provides a way to achieve a similar effect. Figure 3-4 shows a sketch of the Morse potential with a traditional harmonic potential superimposed on top.
Figure 3-4: A sketch of the Morse potential overlayed with a traditional harmonic (symmetric) potential with equally spaced energy intervals, where $x$ is the interatomic spacing. From the National Institute of Standards and Technology [3].

Note a couple important features: (i) The Morse function simulates an anharmonic potential, more reflective of an actual chemical interaction. A truly harmonic (symmetric) potential would not allow for bond breaking or dissociation. (ii) The energy spacings are not constant for the Morse potential and get closer together close to the dissociation energy (where the Morse potential goes flat with increasing interatomic distance), which is also more reflective of an actual interaction. Therefore, the Morse potential will be used to describe the complex Au-S interaction, whose interaction energy rivals those expected of a partially covalent bond. Also, the Morse potential was chosen to describe the C-Au and H-Au interactions, and parameters were used from a previous study done for benzene on gold [50].

**Lennard-Jones Potential:** A well-known and widely utilized potential to simulate Van der Waals interactions is the 12-6 Lennard-Jones (LJ) potential:
\[ E_{LJ} = 4\epsilon[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}] \] (3.11)

For short distances, the 12 term dominates (sharp “hard-sphere”-like repulsion), and for longer distances the 6 term dominates. Two variables \( \epsilon \) and \( \sigma \) must be supplied to LAMMPS here for each LJ interaction.

LAMMPS requires a set of parameters that define every combination of pairwise interactions. However, for LJ interactions, LAMMPS allows the “pair_modify mix” option to be set. This allows one to supply only the values of \( \sigma_{ii} \) and \( \sigma_{ij} \) for each type, and the i,j values are computed according to the mixing type. For this project, arithmetic mixing was used, meaning that the i,j values are

\[
\begin{align*}
\epsilon_{ij} &= \sqrt{\epsilon_{ii}\epsilon_{jj}} \\
\sigma_{ij} &= \frac{\sigma_{ii} + \sigma_{jj}}{2}
\end{align*}
\]

The i,i values were set for C,H,O,S, and the LJ interaction was constructed explicitly for the O-Au interaction.

**Coulombic Potential**: This is the usual Coulomb potential and is accessed in tandem with the LJ potential with the “pair_style lj/cut/coul/cut” style. The potential is of the form

\[ E_C = \frac{Cq_iq_j}{\epsilon r} \]

where \( C \) is an energy conversion constant (not used explicitly), and \( \epsilon \) is the dielectric constant (set to default 1.0). The partial charges on the TTPO molecule were assigned based on the NBO population analysis performed in [4], and are shown in Figure 3-5.
The partial charges are explicitly set in the coordinate (data) file with the atomic coordinates.

**Embedded Atom Model (EAM):** The final interaction potential to consider is the Au-Au interaction, which is modeled by the EAM. This is a successful method for describing metallic systems like those in the gold surfaces in this study. The model was developed in [51] and is based in DFT. The basic idea is that each atom in a solid can be thought of as an impurity lodged in a “host” body, i.e. is embedded in the host. The potential is of the form

\[
E_i = F_\alpha\left(\sum_{j \neq i} \rho_\beta(r_{ij})\right) + \frac{1}{2} \sum_{j \neq i} \phi_{\alpha\beta}(r_{ij}) \quad (3.12)
\]

where \(F_\alpha(...)\) denotes that \(F\) is a function of the electron density \(\rho\) of atom species \(\beta\) at the location of \(i\). \(F\) is the embedding energy, and \(\phi\) is a pairwise potential function between atomic species \(\alpha\) and \(\beta\) that essentially serves as a correction to the energy that takes into account core-core repulsions of the core atoms (so that the main body is treated as non-uniform charge densities).

Here, \(\alpha=\beta=\text{Au}\), and the potential file used is Au_u3.eam, which is distributed with the
LAMMPS package and can be found in the /potentials/ folder. As will be seen, the potential does a good job at reflecting the bulk and surface properties of gold and therefore is a suitable model to describe the systems here.

Table 3.4 displays all (non-Coulombic, summarized above) parameters of the pairwise and bonded/non-bonded parameters used in this study. All energies are in eV, angles in degrees, and distances in Å. The parameters are in accordance with the various pair styles in the LAMMPS manual[41].

<table>
<thead>
<tr>
<th>Interaction</th>
<th>Interaction Type</th>
<th>Value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Au-Au</td>
<td>EAM</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Au-S</td>
<td>Morse</td>
<td>$D_0 = 0.138$, $\alpha = 1.38$, $r_0 = 2.903$</td>
<td>[47]</td>
</tr>
<tr>
<td>Au-C</td>
<td>Morse</td>
<td>$D_0 = 0.0096$, $\alpha = 1.013$, $r_0 = 4.104$</td>
<td>[50]</td>
</tr>
<tr>
<td>Au-H</td>
<td>Morse</td>
<td>$D_0 = 0.0031$, $\alpha = 1.166$, $r_0 = 4.006$</td>
<td>[50]</td>
</tr>
<tr>
<td>Au-O</td>
<td>LJ</td>
<td>$\epsilon = 0.00392$, $\sigma = 2.946$</td>
<td>[52]*</td>
</tr>
<tr>
<td>C-C</td>
<td>LJ</td>
<td>$\epsilon = 0.00372$, $\sigma = 3.399$</td>
<td>[43]</td>
</tr>
<tr>
<td>H-H</td>
<td>LJ</td>
<td>$\epsilon = 0.00065$, $\sigma = 2.599$</td>
<td>[43]</td>
</tr>
<tr>
<td>O-O</td>
<td>LJ</td>
<td>$\epsilon = 0.00910$, $\sigma = 2.959$</td>
<td>[43]</td>
</tr>
<tr>
<td>S-S</td>
<td>LJ</td>
<td>$\epsilon = 0.0108$, $\sigma = 3.563$</td>
<td>[43]</td>
</tr>
<tr>
<td>C=O</td>
<td>Bond</td>
<td>$K_{bond} = 24.72$, $r_0 = 1.229$</td>
<td>[43]</td>
</tr>
<tr>
<td>C-C</td>
<td>Bond</td>
<td>$K_{bond} = 12.579$, $r_0 = 1.55$</td>
<td>[53]</td>
</tr>
<tr>
<td>C-H</td>
<td>Bond</td>
<td>$K_{bond} = 15.914$, $r_0 = 1.08$</td>
<td>[43]</td>
</tr>
<tr>
<td>C=S</td>
<td>Bond</td>
<td>$K_{bond} = 10.563$, $r_0 = 1.79$</td>
<td>[43]</td>
</tr>
<tr>
<td>S-S</td>
<td>Bond</td>
<td>$K_{bond} = 7.198$, $r_0 = 2.038$</td>
<td>[53]</td>
</tr>
<tr>
<td>C=C</td>
<td>Bond</td>
<td>$K_{bond} = 20.340$, $r_0 = 1.400$</td>
<td>[43]</td>
</tr>
<tr>
<td>C-S</td>
<td>Bond</td>
<td>$K_{bond} = 14.262$, $r_0 = 1.675$</td>
<td>[53]</td>
</tr>
<tr>
<td>H-C-C</td>
<td>Angle</td>
<td>$K_{\theta} = 1.51$, $\theta_0 = 120$</td>
<td>[43]</td>
</tr>
<tr>
<td>C-C-C</td>
<td>Angle</td>
<td>$K_{\theta} = 2.73$, $\theta_0 = 120$</td>
<td>[43]</td>
</tr>
<tr>
<td>S-C-C</td>
<td>Angle</td>
<td>$K_{\theta} = 2.69$, $\theta_0 = 120$</td>
<td>[53]</td>
</tr>
<tr>
<td>O-C-C</td>
<td>Angle</td>
<td>$K_{\theta} = 3.11$, $\theta_0 = 120$</td>
<td>[53]</td>
</tr>
<tr>
<td>H-C-C-C-C</td>
<td>Dihedral</td>
<td>$K_D = 0.157$, $d = -1$, $n = 2$</td>
<td>[43]</td>
</tr>
<tr>
<td>H-C-C-H</td>
<td>Dihedral</td>
<td>$K_D = 0.157$, $d = -1$, $n = 2$</td>
<td>[43]</td>
</tr>
<tr>
<td>C-C-C-C-C</td>
<td>Dihedral</td>
<td>$K_D = 0.157$, $d = -1$, $n = 2$</td>
<td>[43]</td>
</tr>
<tr>
<td>O-C-C-C-C</td>
<td>Dihedral</td>
<td>$K_D = 0.130$, $d = -1$, $n = 2$</td>
<td>[53]</td>
</tr>
<tr>
<td>S-C-C-C-C</td>
<td>Dihedral</td>
<td>$K_D = 0.157$, $d = -1$, $n = 2$</td>
<td>[53]</td>
</tr>
</tbody>
</table>

*—Computed manually from Au-Au and O-O parameters from mixing rules
3.3 Visualization and Input/Data File Generation

3.3.1 PWgui

There are many input parameters in QE, and generating an input file in a vim editor from scratch can be a difficult task, and some variables can easily be left out. PWgui is a graphical user interface that allows for the easy generation of input files for the pw.x executable (which can be then easily manipulated and adapted to a CP calculation), and is an option for download with the QE package from the developers’ portal QE-forge [54]. PWgui has all of the cards for the calculations chosen displayed, and all of the variables within each card are neatly organized and have a help button that gives details on the variable. It is an easy way to generate an input file and visualize all available variables for each calculation, and one doesn’t have to worry about where the variables go, which isn’t the case for a vim editor. After gaining experience with input files, it will likely be much quicker and simpler to edit the input file directly in the command window.

3.3.2 XCrysDen

Output data from QE and LAMMPS would be of little interest or help without the ability to visualize the results from the calculations. In addition, it is important to be able to see the input structure to check for deformities. The software X-window CRYstalline Structures and DENsities (XCrysDen) allows for the easy visualization of input and output data using files for QE. The program accepts pw.x and ep.x input files output files, whether it is a structural calculation or dynamics simulation. With regards to LAMMPS, XCrysDen can be utilized to analyze .xyz or other more common file formats. It does not recognize .lammpstrj files (the output extension for a LAMMPS trajectory). However, LAMMPS enables one to output
an .xyz coordinate file every N timesteps so that coordinates, lengths, bonds, etc. can be calculated.

When designing a calculation or simulation to be run, XCrysDen will first be useful in order to visualize the unit cell constructed in the input file (either pw.x or cp.x input files for QE or .xyz coordinate files for LAMMPS). It is vital that the unit cell be built such that there are no overlapping atoms, and that the interatomic distances are what they should be. By visualizing the input file in XCrysDen, one can verify that the atomic positions are correct. This can be done qualitatively by visual inspection, and one can also use the distance calculator in XCrysDen, or the angle calculator to verify that all values are precisely correct. It is essential to visualize the unit cell before any calculation, structural or dynamical. If the visualization is not able to be generated by XCrysDen, it likely means that there is an error or errors in the atomic coordinates in the input file.

**Note:** It is not essential to fill out all possible input variables in order to visualize the unit cell in XCrysDen for any simulation. The unit cell can be visualized with the atomic coordinates and other basic information about the cell parameters and calculation type.

In addition to generating a 3-D rotatable image, XCrysDen has the ability to make animations for MD simulations. This can be done in cp.x with the .cppp (CP post-processing) executable for QE to generate an Animated XCrysDen Structure File (.axsf), or in LAMMPS by outputting the .xyz coordinate files (the .xyz files are placed after each other, similar to a flip book).
3.3.3 VMD

Visual Molecular Dynamics (VMD)[55] is a powerful MD visualization software from the Theoretical and Computational Biophysics Group at the University of Illinois, Urbana-Champaign. In comparison to XCrysDen, it has many advantages. It has the ability to easily load and manipulate unit cells containing thousands of atoms, and runs the trajectories of said unit cells effortlessly (whereas XCrysDen tends to take a long time to load trajectories of a few thousand timesteps and runs slowly). In addition, VMD has several advanced options for lighting, atom representation (spheres, lines, dots, etc.) and colors, navigating with the mouse and keyboard (so up-close inspection of atoms can be performed), overlaying molecules, and more. While VMD was much better at processing and viewing trajectories and large unit cells, XCrysDen was typically easier to view atom distances, coordinates, and angles.

3.3.4 TopoTools

Recent versions of VMD come pre-installed with a useful plug-in called TopoTools[56]. It is used for relatively easy generation and manipulation of LAMMPS data files. It can be used through the Tk console in VMD (“Extensions → Tk Console” in the VMD Main window). While it has several useful properties for topology generation, the most useful here was the “writelammpsdata” command to generate a LAMMPS data file. An .xyz structure file was loaded into VMD that contains the atomic coordinates for the atoms in the unit cell, and then the “writelammpsdata” generates a LAMMPS data file in the format recognized by LAMMPS. This command does not set the unit cell boundaries, interaction parameters, or other details about a simulation. This is done either later in the data file or in a LAMMPS
3.3.5 Moltemplate

While TopoTools is a valuable tool that has the capabilities to manipulate topology and interaction information, Moltemplate\cite{57} was used to manipulate and add molecules, substrate layers, etc. Moltemplate was used because it easily allows one to define a self-contained molecule (containing coordinate, bond, angle, dihedral, and charge information), and construct a unit cell piece-by-piece. For instance, a slab of gold was created that consists of three layers of gold. After, it is a simple matter to tile the slab next to each other as many times as desired to increase the surface size. The same is true for a TTPO molecule. Once the molecule is created, it can be copied, rotated, and moved with one command so that several molecules can be placed on a surface easily.

Moltemplate contains its own file type (.lt) in which it uses to generate LAMMPS data files. After a LAMMPS data file is created (from TopoTools as described above), the ltemplify.py Python tool in Moltemplate can create an .lt file “MyMolecule.lt” from the data file generated in TopoTools “MyDataFile.data” with the command

\texttt{ltemplify.py -name MyMolecule MyDataFile.data > MyMolecule.lt}

Now the molecule “MyMolecule” (a slab of gold, a TTPO molecule) can be copied, rotated, etc. by creating another .lt file for the system, say “System.lt”. This is where the pieces will be put together to construct the data file for the system to be used in LAMMPS. First, one must import all “Molecule.lt” files that are going to be used with the command
import “MyMolecule1.lt”
import “MyMolecule2.lt”

for all species to be used. Here, for example, the top of a system.lt file may look like

import “TTPO.lt”
import “3LayerAu.lt”

And then each object is placed by adding the line (for example) “TTPO1 = new TTPO”. To move or rotate a molecule, the command is altered to

“TTPO1 = new TTPO.move(x,y,z).rot(θ,x₀,y₀,z₀)”, which will first translate the molecule by x,y,z Angstroms, then rotate it θ degrees about an axis that passes through the origin and the point x₀, y₀, z₀. Advanced options for manipulation can be found in the Moltemplate manual [58]. From here, one final step is taken to generate the LAMMPS data file to be used. The command is:

moltemplate.sh -atomstyle “full” -nocheck -vmd System.lt

This does two things: (i) It generates a file System.data that can be then used as a data file in LAMMPS (if no further alterations are desired) and (ii) The -vmd flag opens the system in VMD to inspect the topology. The -nocheck flags tells the program not to check for force field information (this was prepared in a separate input file). See Appendix A for a step-by-step guide for LAMMPS data file generation.
3.4 Machines

3.4.1 Vortex

The magnitude of complexity involved in both molecular structure and dynamics simulations requires sophisticated and robust machines to do the calculations. Input files, visualization, and batch submission files were prepared locally (on a Dell Studio 17 laptop), then submitted to Vortex, an 8-node, 64 processor cluster with 192 GB of memory that is housed in DeMeritt Hall at UNH. The multiple nodes of Vortex allow it to perform calculations that would be unrealistic or impossible on ordinary desktop computers.

SLURM

Vortex uses a workload manager called Simple Linux Utility for Resource Management (SLURM). SLURM provides an easy environment to submit jobs to the queue in Vortex. The way SLURM prepares a job to be completed is through a batch submission script. Figure 3-6 shows an example of a submission script for a PWscf calculation.

```
#!/bin/sh
export OMP_NUM_THREADS=4
mpirun pw.x-mp -npool 1 < 1Au4diprelax2.pw.in > 1Au4diprelax2.pw.out
```

Figure 3-6: Example of a batch submission script in SLURM.

The first line declares the script as a bash shell script, a Unix command language. The second line sets the maximum number of threads per node the job is to run. The third line contains information about the program that runs the job, the number of pools, and the input and output files. “mpirun” is the executable that launches jobs within SLURM. “pw.x-mp” says that the calculation is in the pw.x package, and it is to be run in parallel.
(with the executable mpirun). “-npool 1” declares that the number of pools per thread is 1
(the number of pools should be a multiple of the number of k-points in the calculation for
optimal parallel performance). Finally, the angled brackets encloses the input file for pw.x,
and the last term is the output file (created when batch script is submitted for execution).

3.4.2 Trillian

Larger calculations require even more processing power, which, in turn, require even larger
machines than Vortex. Therefore, the primary machine used in this project was a 132-node
(4,224 core) Cray XE6m-200 supercomputer, “Trillian” [59]. Trillian was acquired through
an NSF Major Research Instrumentation grant, and is housed in Morse Hall at UNH. It
offers several useful features such as a module system that allows computing environments
to be set up by easily loading various modules. See Appendix B for commands for general
comments on Trillian and useful commands on Trillian. While using Trillian is, in general,
less straightforward than other machines, it offers great benefits in terms of computational
ability. The calculations in this project were often ran on dozens of nodes, so employing
Trillian was necessary to obtaining many of the results reported in this study.

PBS

Similar to how Vortex uses SLURM to schedule its jobs, Trillian uses a software called
Portable Batch System (PBS). While the general idea is similar to SLURM, PBS uses different
commands for job submission and has a different anatomy for a batch submission script.
Figure 3-7 shows an example for a batch script for a job in a PBS environment.
Figure 3-7: Example of a batch submission script in PBS.

The top section contains variables preceded by the #PBS argument and define the maximum compute time (wall time), number of nodes (mppwidth), processors per node (mppnppn), and log file to output error and job output to. The next line tells the program where to find the input file for the job (here, "${HOME}/lammps_calc"). This is followed by the depth (OMP_NUM_THREADS), and the aprun command, which is the command to be used on Trillian to schedule jobs on the compute nodes. This line contains the executable to be used, see Appendix C for a description of how each of the executables used in this project were built. Submission of a job from the command line is done through the command “qsub <my_batch_script>”, where it is then submitted to the queue and will be completed once enough resources are free to accommodate the number requested.
4.1 \( \pi \)-Conjugation

A particular challenge in the field of organic semiconductor production is finding candidates with a charge carrier mobility comparable to that of silicon (the leading candidate in efficiencies for photovoltaic devices), whose amorphous form has a charge carrier mobility of 1-10 cm\(^2\) V\(^{-1}\) s\(^{-1}\) [60]. Molecular orbital (MO) theory states that electrons become delocalized in conjugated systems (systems with alternating single and double bonds), which helps to increase stability, symmetry, and charge-carrying ability [61]. Systems of planar organic molecules demonstrate \( p \)-orbital overlap and utilize this efficient electron mobility. There are different ways a planar molecule can stack on a surface, two of which are demonstrated in Figure 4-1. Lamellar stacking is considered to be superior in terms of charge-carrying ability, due to intermolecular \( \pi - \pi \) overlap.
However, organic crystals contain significantly weaker intermolecular interactions than inorganic solids [62]. This can increase intermolecular distance, detracting from conduction ability. So, the problem persists to find molecules who strike a balance between preferred planar stacking while preserving close-packed distances to remain highly conductive. TTPO assembles in a head-to-tail stacking formation in its crystalline form on surfaces, achieving the desired $\pi$-orbital overlap [4]. This stacking formation, among other characteristics, makes TTPO a promising organic semiconducting candidate.

4.1.1 Angular Assembly

Pure pentacene on flat Au(111) surfaces assumes a planar geometry parallel to the surface [63, 64]. The pentacene derivative 6,13-dichloropentacene (DCP) also prefers to assemble flat on Au(111) [65]. This, perhaps, isn’t too surprising, since both pentacene and DCP are symmetric molecules about the long and short axes, therefore a parallel assembly on interacting metal surfaces might be expected. However, TTPO is unique in that it is a
stable, asymmetric (with respect to the long axis) molecule and adsorbs on metal surfaces at an angle:

Figure 4-2: Pentacene and two of its derivatives isolated on Au(111). TTPO is unique because the sulfur end tilts towards the surface.

The angular assembly of TTPO is attractive due to increased molecular overlap. This, along with its inherent polarity, makes TTPO a promising candidate for nanoelectronic applications.

4.2 Solid-State Surface Diffusivity

Isolated TTPO molecules do not remain stationary on Au(111) surfaces and Au(788) terraces. Scanning Tunneling Microscopy (STM) images have shown that, at room temperature, TTPO diffuses across the gold surface (Figure 4-3).
Surface diffusion is not unique to TTPO. Pentacene has been seen to diffuse on gold and contaminated and sputter-cleaned SiO₂ surfaces [66]. Even large planar organic molecules like 4-trans-2-(pyrid-4-yl-vinyl) benzoic acid (PVBA) have been seen via STM to be mobile on SiO₂ and Pd(110) surfaces [67]. Diffusion is a necessary process for bottom-up self-assembly to occur for molecular systems, as molecules need to move across a surface to position themselves in an organized manner. However, without any type of anchoring, the self-assembly process cannot be initiated. Molecules will continue to diffuse until a low-energy preferential site is found for the Au-S interaction to dominate the mobility of the molecule. There needs to be a controlled way to anchor a molecule so that it can serve as a
Several methods exist to study surface architectures, including STM, Low-Energy Electron Microscopy (LEEM), Auger Electron Spectroscopy (AES), Atomic Force Microscopy (AFM), etc. These methods are highly sensitive, as the distances being observed are very small (often on the order of Angstroms for atoms and molecules), and the timescales in which events occur at the molecular level are also very small (non-trivial behavior can occur in femtoseconds). Therefore, diffusing molecules are difficult to study with these visualization methods, and computational simulations must be employed to study the details of diffusion across a surface.

4.2.1 Diffusion Coefficient from Langevin Dynamics

Section 2.4.1 proposed a method of thermostatting a system via placing the system in a fictitious solvent, where the stochastic stimulations to the atoms cancel out over time. For a gold substrate at a finite temperature, atoms in the surface layers will be vibrating, which can provide an additional amount of energy in an uncorrelated (i.e. random) direction [68]. Said another way, the adsorbed molecules are effectively coupled to a heat bath when the system is thermostatted to a finite and sufficiently high temperature (so that there is a significant oscillation amplitude of the substrate atoms). Although the substrate is in solid form, the finite temperature for a large surface leads to an effectively uncorrelated and hence random vibrational surface. Therefore, the molecules on the surface can be regarded as coupled to a bath, the surface, in which it receives energy and subsequently returns it. So, the Langevin equation is often used for descriptions of molecular diffusion on surfaces.

The Langevin equation couples nicely with interatomic potentials to provide a convenient and informative way to describe molecular dynamics. When studying diffusion, one needs a
specific way to quantify the motion of the molecules under study. If one pictures a lone atom undergoing Brownian Motion on a surface, the mean displacement averaged over time would be approximately zero (the same is true for larger molecules). This is because the Langevin Equation enforces a random “kick” to each atom to simulate a desired temperature, so, over time, these random fluctuations cancel out. However, if one considers the mean square displacement, i.e. the sum of $dx^2$, $dy^2$, and $dz^2$, then the value would increase over time.

If one multiplies both sides of Equation 2.26 (Langevin Equation) by $\bar{x}(t) = x(t) - x(0)$ and averages over the ensemble

$$m\langle \bar{x} \frac{\partial \bar{x}}{\partial t} \rangle = -\gamma \langle \bar{x} \ddot{x} \rangle + \langle \bar{x} \dot{F}(t) \rangle$$  \hspace{1cm} (4.1)$$

(Here an ensemble average is denoted as $\langle A \rangle$ and is defined as the average over phase space for a system:

$$\langle A \rangle = \frac{\int A e^{-\beta H(q_i,p_i)}}{Z}$$

for a system Hamiltonian $H$ and partition function $Z$), the condition in Equation 2.27 can be employed to give a first-order PDE for $\langle \bar{x} \ddot{x} \rangle$:

$$m \frac{\partial \langle \bar{x} \ddot{x} \rangle}{\partial t} = \gamma \langle \bar{x} \ddot{x} \rangle + 3k_B T$$  \hspace{1cm} (4.2)$$

For the last term, the equipartition value was inserted for $m\langle \dot{x}^2 \rangle$. Solving this PDE gives a formula for the mean squared displacement of an atom in a simulation

$$\langle [\bar{x}(t) - \bar{x}(0)]^2 \rangle = \frac{6k_B T}{\gamma^2} (\gamma t - m(1 - e^{-\gamma t/m}))$$  \hspace{1cm} (4.3)$$
Equation 4.3 has two immediately useful regimes. First, for small times \( t \ll \gamma^{-1} m \), the expression can be Taylor expanded to yield

\[
\langle [\vec{x}(t) - \vec{x}(0)]^2 \rangle \simeq \frac{3k_B T}{m} t^2
\]  

\( (4.4) \)

So, for small times in a Langevin scheme, the mean squared displacement has a quadratic dependence on time. This is the regime in which the system is not yet equilibrated to a steady-state temperature. More interesting is the long time regime, \( t \gg \gamma^{-1} m \):

\[
\langle [\vec{x}(t) - \vec{x}(0)]^2 \rangle \simeq \frac{6k_B T}{\gamma} t
\]  

\( (4.5) \)

At long times, the mean square displacement is linearly proportional to time, where the proportionality constant \( \frac{k_B T}{\gamma} = D \), is the diffusion coefficient. The diffusion coefficient is unique for a given solute (or adsorbed molecule in the case of this study), in a given solvent (or surface), at a certain temperature. Therefore, it is truly unique to each chemical environment. The MSD is also proportional to the diffusion coefficient when thermostatted via a NH thermostat (\( MSD = 6Dt \)). **Note:** The above derivation describes 3D diffusion, hence the factor of 6 in Equation 4.5. However, surface diffusion is generally a 2D process, so it is a better approximation to replace the 6 with a 4 when analysing diffusion coefficients, see Chapter 5. Measuring diffusion coefficients gives insight into a molecule’s mobility in a given chemical setup, which is the underlying premise for molecular self-assembly. The Langevin Equation describes random fluctuations in the trajectory, and diffusion coefficients can provide a way to quantify the relatively long-term motion of the particles and can give insight into the ways these molecules come together to form ordered structures.
4.2.2 Arrhenius Plots and Diffusion Barrier Energy

Surface diffusion is, in part, a thermally activated process. At low temperatures, molecule-substrate interactions dominate the intramolecular thermal vibrations, causing molecules to remain stationary. However, it is not hard to picture increasing the temperature (and therefore thermal energy $\sim k_B T$), and helping to give atoms/ molecules sufficient energy to hop from one local minimum of potential to another. Since the thermal energy is often much lower than the diffusion barriers needed for adatoms and molecules on the surface [69], it is expected that thermal energy alone will not supply enough energy to diffuse. However, temperature dependence of kinetic processes still give insight to the energetics involved in diffusive processes.

The energy landscape of a smooth surface like a wide, flat Au terrace can be thought of as a series of potential wells, shown in Figure 4-4. In practice, this is an obvious oversimplification: the exterior shell of valence electrons on a gold surface do not represent spherical cavities, but consist of a complex map of orbitals in which electrons are most likely to be found. However, by reducing the idea of diffusion to this "hill and valley" model, valuable insight can be gained into the kinetics of TTPO diffusion through classical MD simulations.
Figure 4-4: Simplified graphic of the potential energy surface of a metal surface. In (a), two adatoms are shown in a stable state (position 1, in a bridge or well site), and a metastable state (position 2, on top of a surface atom). (b) shows a theoretical map of potential energy of a single well, which is modelled as an asymmetric well. Finally, part (c) demonstrates the concatenation of several potential wells which models the metal surface and the diffusion landscape. Adapted from [7].

Surface diffusion is a more complicated process for a robust molecule like TTPO. Whereas for a single adatom there is a sole interaction energy to consider (atom-substrate), TTPO has several interactions that are working in tandem to contribute to adsorption on the surface: the (relatively) weak Van der Waals attraction from the pentacene backbone and oxygen atom to the gold atoms, and the strong S-Au partially-covalent bond. However, the molecule
will still seek local minima along the surface, and there is good reason to suspect that the local minima could be adjacent or nearby bridge or hollow sites. Therefore, an activation energy barrier exists for this thermally activated process to occur, the details of which can be extracted from the Arrhenius Equation. This is an equation that models the temperature dependence of a kinetic constant in a chemical reaction. In the present case, it can be used to extract the activation energy for diffusion by considering the diffusion coefficients from the methods described above. The Arrhenius Equation to be used here is

\[
D(T) = D_0 e^{-E_a/k_B T}
\]

where \( D \) is the diffusion coefficient, \( D_0 \) is a pre-exponential factor, and \( E_a \) is the activation energy barrier for diffusion. By plotting the logarithm of \( D \) vs. \( 1/k_B T \), the slope will be equal to \(-E_a\). By analysing the precise dependence of \( D \) on \( T \), one can get a better understanding of the energetics of TTPO surface diffusion on gold.

### 4.2.3 Substrate Dependence and Au(111) Reconstruction

The details of a molecule’s diffusion across a surface greatly depends on the substrate on which the molecule is placed. There have been studies that show that molecules tend to prefer a direction of diffusion that directly relates to the Miller indices of the surface. In [67], the PVBA molecule displacement was observed to be exclusively along the [1\,\bar{1}\,0] direction when it was placed on Pd(110). This suggests a dependence on primitive lattice directions on surface diffusion, and that anisotropy in a substrate can possibly give insight into the details of diffusion.

Another critical factor in surface absorption and diffusion is surface reconstruction. In
a bulk crystal, forces on a given atom are balanced in all directions. However, a crystal (in the case of this study, face-centered-cubic (fcc) Au) can be cleaved in different directions to expose a variety of surface configurations, each with its own set of minimized energies, molecular spacings, and potential to chemisorb atoms and/ or molecules. Of particular interest in this study is when bulk Au is cut along the plane that intersects the a,b, and c directions at the same point to yield Au(111), see Figure 4-5.

(a) Top-down view showing hexagonal formation of atoms. Offset layers are indicated by the A, B, and C labels.

(b) Plane showing the cut direction to obtain (111) crystal.

(c) Top-down view of the close-packed formation with the sixfold symmetry indicated by the dashed lines.

Figure 4-5: Different representations of fcc (111) surface and close-packed formations. From [8].

Au(111) is a highly favorable surface for adsorption. It is atomically smooth with no bumps, and was found to have a relatively low unreconstructed surface energy of 1.52 $J/m^2$ [70]. However, the Au(111) surface reconstructs into a herringbone ($22 \times \sqrt{3}$) formation (in Wood’s notation) in order for the surface atoms to achieve their lowest energy states, a quality that isn’t observed for all fcc (111) metals. The herringbone pattern is characterized by striated alternating hexagonal close packed (hcp) and fcc layers, and occurs spontaneously at room temperature [9]. The average Au-Au distance after reconstruction was measured to be $2.82 \pm 0.25\AA$ [9]. This defect of the surface is essentially due to the uppermost layer
having to accommodate more atoms, and leads to a contraction of the top layer by about 4% with respect to the atoms in bulk [71], and causes the 120° zig-zag formation seen in Figure 4-6.

![Image](image1.png) (a) Image showing the zig-zag pattern characteristic of herringbone reconstruction for Au(111). Two different types of elbows are shown. X-type elbows have a point dislocation, and y-type contain no dislocation. Pinch-in dislocations refer to the dislocations that contain a missing atom, and pinch-out contain an additional atom. Note the alternating fcc and hcp patterns separated by inhomogeneous elbows (bright lines).

![Image](image2.png) (b) High-resolution image of pinch-in dislocation (enclosed by hexagon).

Figure 4-6: STM images showing herringbone reconstruction characteristics. From [9].

**Vacancy defects**

One of the reasons the reconstructed Au(111) surface is interesting is that the pinch-in defect (vacant atom) site is essentially a low-energy well, which can potentially attract atoms and perhaps anchor them down, therefore keeping them localized. This is critical, as mentioned above, for self-assembly because it could serve as a means to initialize the molecular conducting chains. Many studies exist characterizing diffusion of adatoms on various transition metal surfaces [72] [73]. The natural periodicity of a pristine surface does not provide a
preferred site for stationary adsorption, and often, even at room temperature, molecules and adatoms are unable to remain fixed to a single location [6] [73]. Although adatoms are often easily trapped in a vacancy, there is reason to believe that the defects might have an effect on a robust molecule like TTPO. Ground-state DFT calculations in [6] and in this study have shown that TTPO adsorbs with the central sulfur 2.5-2.7 Å directly above a gold atom, so it will be significant to see the effects of a vacancy site or multiple vacancies on the ability of this strong Au-S interaction to overcome its diffusive nature.

**High-Index Cuts: Vicinal Surfaces**

Unreconstructed Au(111) is a large, flat, smooth substrate which, ideally, is free of any bumps, ridges, or any breaks in periodicity. As discussed above, this is a hindrance for self-assembly, as there is no basis for molecular chains to form. However, higher-index planes of surfaces have a natural aperiodicity in the form of steps. Au(788) is a stable high-index surface of the gold crystal that has 3.83 nm wide steps that have 16 atomic rows per terrace, and is formed by cleaving Au(111) by 3.51° in the [211] direction [10]. It has \{111\}-like step edges that are one atom high, see Figure 4-7.

![Figure 4-7: Two types of stable vicinal Au(111) surfaces. On the left, the steps are \{111\}-like (triangular packing). On the right, they are \{100\}-like (square packing). The terraces retain the 111 pattern. From [10].](image-url)
The idea that step edges of fcc transition metals could provide a favored adsorption site for chain or thin-film growth is a widely explored topic in surface science. For example, growth studies for carbon nanofibers has been studied on the flat Ni(111) surface and step edge [74]. In the study, it was calculated that the binding energy of \( C_2H_2 \) at the step edge was 0.7 eV higher than on the flat surface. This indicates that a step edge can be a favored site for a TTPO to bind to, which can serve as an anchor for stable chain formation.

Au(788) surfaces are of interest in this study because they have been sites of observed TTPO chain formation via STM [6]. TTPO chains were seen on the terraces, with the chain anchored to the step edge. The stable configuration of TTPO chains seen in Figure 4-8 is of particular interest because it is a verified case of TTPO forming an ordered assembly in a controlled environment. This is precisely the behavior desired in directed self-assembly, and the observation warrants the current study of how TTPO forms these ordered structures.

Figure 4-8: High-resolution STM image showing the ordered chain structure of TTPO on Au(788) steps. From [6].
Figure 4-8 shows the highly-ordered chain substructures that can be resolved in the direction of the steps, with a lateral offset structure within chains. The image suggests that Au(788) provides a framework for ordered molecular substructures to assemble in a stable fashion. Therefore, it warrants investigation into how TTPO forms these ordered chains, and the effect of varied temperature on anchoring ability.
CHAPTER 5

RESULTS

5.1 Comparison of DFT and Classical Mechanics Results

Due to their large computational effort, ab initio calculations were primarily used as a basis for verification that the faster, more efficient classical dynamics and structural calculations performed in LAMMPS were well-parameterized. Since calculations in QE are from first principles, they were considered to be the model in which to strive for when constructing the classical simulations. Results for various physical setups in QE are summarized here, and are followed by the analogous results in LAMMPS.

5.1.1 Tilt Angle

One critical aspect of TTPO assembly on gold that can be studied and compared to past results is the angular conformation. A DFT energy minimization was ran in Quantum ESPRESSO for a single molecule on four layers of Au(111). The gold atoms were held fixed, and the initial orientation of the TTPO molecule was 4 Å above the surface, tilted about the x-axis by 45 degrees. This was arbitrarily chosen as a system far from equilibrium but
not so far as to not adsorb onto the surface. The calculation was ran with an ionic energy threshold of $10^{-4}$ Ry, a force threshold of $10^{-3}$ (atomic units a.u.), and a self-consistency threshold of $10^{-6}$. The tilt angle of a single molecule on 4 layers of close-packed gold via DFT was calculated to be $11.5^\circ$. This is consistent with previous DFT results reported in [6].

![DFT relax result for a single molecule on close-packed 4-layer gold. The tilt angle is 11.5°, and the difference in height between the central sulfur atom and the top layer of Au is 2.5Å](image1)

![Top view of DFT relax results. The sulfur bridge aligns itself so that the central sulfur atom is directly above a gold atom.](image2)

Figure 5-1: DFT relax results. Periodic boundary conditions were enforced in all three directions, and the kinetic energy cutoff for wavefunctions was 30.0 Ry.

In addition to the tilt angle, the distance in z coordinates for the central sulfur atom and the topmost layer of gold is 2.5Å. The molecule stayed rigid, with there being a slight bending of the outer sulfur atoms inward. Also, there is a small rotation about the z axis (Figure 1(b), out of the page). This is likely dominated by the sulfur atoms, whose strong attraction to the gold finds a minimum potential energy with this small rotation angle.

An energy minimization calculation was run in LAMMPS in the same physical conditions and identical initial conditions as the above DFT result. The results are shown in Figure 2. The results show a well-parameterized system for TTPO on gold. The minimization
was split into two parts: an initial damped dynamics run of 500,000 timesteps (min_style quickmin) with a 10ps timestep to drain most of the kinetic energy from the system, followed by a conjugate gradient run with the quadratic option to probe the local energy field for a minimum. The latter was ran until the linesearch alpha value in the algorithm was zero, indicating that a movement that would further minimize the system’s energy could not be found (to machine precision). This scheme was chosen because damped dynamics slows down as the system approaches convergence, so splitting the calculation into two parts helped reach convergence.

(a) LAMMPS relax result for a single molecule on close-packed 4-layer gold. The tilt angle is 14.1° (slightly higher than DFT result), and the difference in height between the central sulfur atom and the top layer of Au is 2.3Å (slightly closer than DFT result).

(b) Top view of LAMMPS relax results. Contrary to the DFT result, the middle sulfur now sits directly above a hollow site between gold atoms.

Figure 5-2: Classical mechanics relax results. Physical environment and initial configuration were the same as for DFT.

The central S-Au top layer distance is 2.3Å, 8% smaller than the DFT result. This makes sense, since the middle of the sulfur bridge now sits in a hollow site, so it is occupying an area of relatively low electron density, therefore it is slightly closer to the surface. Similarly to the DFT relax, the molecule retained its rigidity and the outer sulfurs now bend in slightly more towards the center atom. The tilt angle increased slightly to 14.1°, which could be
from the fact that the sulfur is closer towards the surface, angling the molecule downward more.

From direct comparison of DFT and classical-based relaxations on a flat surface, it leads one to assume that the force field constructed in LAMMPS is well-parameterized. As a further check, we can compare the results at an Au(788) step edge. Figure 3 shows a DFT relaxation result computed in [6] for a molecule at the step edge. Tilt angles are shown in the diagram, and are 16.7° for the tilted-away from the edge case, and 16.8° for a molecule that initially hangs over the step edge.

Compare the above results to those shown in Figure 4, an energy minimization in LAMMPS. For this relax, no damped dynamics were used (since the molecule in each case was initially placed close to equilibrium, a simple conjugate gradient run was enough to find a true local minimum). The facing-away molecule was placed at an arbitrary angle (greater than 0 degrees, less than 90 degrees), and the step-facing molecule was tilted essentially flat above the below terrace.

Figure 5-3: DFT relax results for one molecule at the step edge, performed in [6]. Angles of minimum energy are shown for the step-facing and facing-away configurations.
Figure 5-4: LAMMPS relax results for one molecule at the step edge. Angles for the facing-away and step-facing assemblies with respect to the solid line are 9.3° and 12.9°, respectively.

The results show that both molecules stay anchored to the edge by the sulfur bridge and rotate about this fixed axis, as was the case in the DFT result. Also, the molecules still stay placed with the bridge directly (or nearly close to) over the row of gold atoms at the step edge, which was true in the DFT case as well. However, the angle for the facing-away configuration is decreased to 9.3° degrees, as opposed to the 16.7° seen above, and decreased to 12.9° for the facing-away configuration, as opposed to 16.8° in the quantum-based calculation. It is difficult to say what could be causing this discrepancy, but the important behavior of the molecule is retained, most importantly its binding ability to the step edge and its preference to stay at the step edge. This is the critical aspect of TTPO on Au(788) that allows for ordered structures to form, as will be shown. Therefore, the model constructed in LAMMPS still serves as a viable means of investigation for surface diffusion and self-assembly of TTPO on gold.

As a final comparison of tilt angle, a small test dynamics simulation was performed in both the DFT and classical frameworks. An MD calculation was performed for a single molecule on the same small test system (one molecule on three-layer slab of gold, periodic boundary conditions in three Cartesian directions) for 5 ps. The results are displayed in Figure 5 for t=0.415 ps, t=1.9 ps, t= 3.6 ps, and t=5 ps. There are a couple of things to note. 1: This
is a very small dynamics unit cell and time. It was chosen purely as a basis for comparison. 2: The molecules’ trajectories did not overlap at every timestep, this wouldn’t be expected for two calculations from very different methodologies. However, it is remarkable how close the trajectories were for being calculated in two very different fashions. The molecules both start flat with precisely identical initial conditions, then the oxygen rises up to achieve the maximum angular tilt seen in the 0.415 ps snapshots (during the temperature equilibration phase), then there is a flattening of the molecule with oscillations about a lower angular tilt afterwards. There were small rotations about the z axis in the classical case, but it rotated back after a short time.
Figure 5-5: Small comparative molecular dynamics simulation ran in the DFT and classical-mechanics based frameworks.
5.1.2 Close-Packed Direction Alignment

While surface diffusion of a molecule on thermalized metal tends to be in a random direction on the short-scale (since the molecule gets random kicks from the metal heat bath), Figure 4-5(c) shows that the crystal structure of an underlying substrate may eliminate total isotropy and create preferential directions for diffusion. The surface of the Au(111) substrate studied here has sixfold symmetry, indicated by the dashed lines in 4-5(c). So, it is worth investing the effects of these close-packed directions on the diffusion of a large molecule like TTPO.

To analyse the effects of these close-packed directions on preferential site hopping or alignment of the molecule, a small DFT and a large classical simulation were prepared. The DFT calculation, naturally, had a small unit cell: Two molecules of TTPO, placed on one layer of gold (held fixed) that was six atoms wide, and ten atoms long. The initial purpose of this simulation was to see how two molecules interacted in the DFT framework. However, the results showed something interesting: while the molecules didn’t seem to interact on the timescale probed (about 5 ps), one of the molecules did perfectly align its sulfur bridge in along one of the symmetric directions, and then diffused approximately one lattice site along the direction corresponding to the plane of the symmetry direction. See Figure 5-6.
Figure 5-6: A small DFT simulation showing a clear preference for alignment of the sulfur bridge along a close-packed direction. (a) The molecule lowered itself to the surface from the initial placement and raised its temperature to 300K. (b) The molecule has clearly rotated 60° clockwise to align the sulfur bridge along a close-packed direction, indicated by the red arrow. (c) The molecule moved about one lattice site along the direction of the blue arrow, which is the plane associated with the close-packed symmetry direction. The other molecule can be seen to begin rotating towards the direction as well.

While the above result was for a stationary gold surface that was only one layer, the surprising thing is that the same phenomenon was seen in the classical simulations with five layers of gold, three of which were allowed to move. The unit cell consisted of 10,512 atoms (4 TTPO molecules). The four molecules were placed far apart on a large gold slab, so effects from the surface alone could be isolated. Since surface gold atoms were now allowed to move in all three Cartesian directions, more chaotic behavior was seen than the simple alignment seen in Figure 5-6. However, the molecule still showed a definite preference for aligning its sulfur bridge along a close-packed direction.
Figure 5-7: LAMMPS simulation at 300K showing an isolated TTPO molecule travelling along a close-packed symmetry direction. The molecule travelled approximately two lattice sites (twice as far than in the above DFT simulation), and it maintained its alignment in the symmetry direction. The blue arrow in (b) indicates the direction of motion along the symmetry plane, and the red lines show the sixfold symmetry for reference.

Naturally, the molecule did not maintain perfect alignment along this direction (as can be seen by the slight twist in Figure 5-7(a)). The thermal motion of the substrate caused the molecule to be relatively unstable at a local minimum site, and alignment in even the close-packed symmetry direction was short-lived. Also, not every diffusive event was of this fashion; Sometimes molecules would diffuse via a completely different process. However, despite the chaotic behavior, the molecules tended to show a definite preference for wanting to align their sulfur atoms along these symmetry directions. There were multiple cases of mobility along these directions, even at room temperature. This is a fascinating result as it was seen in both the DFT and classical frameworks and shows that while the diffusion process does not obey one definite mode, the anisotropy of the substrate has a definite effect on mobility.
All of these results confirm that while there are noticeable differences in the trajectories of the two frameworks, the LAMMPS framework is suitable for studying the TTPO-Au systems with only minimal sacrifice of accuracy for structural analysis. The benefit of being able to study large systems for long timescales outweighs the minimal accuracy lost, especially since mainly structural analysis is sought here, as opposed to energetics, in which DFT analysis would likely be inevitable. Therefore, the remaining results are for a purely classical framework.

5.2 Stable Chain Formation at Step Edges at 373K

As described in Chapter 4, Au(788) step edges are an experimentally verified source [6] of stabilizing mobile TTPO molecules at room temperature, and serving as a template for chain formation. Here, this phenomenon was studied via simulation to see the chains form from a random, relatively low monolayer coverage of TTPO on Au(788) terraces. A low coverage of TTPO (36 molecules) were placed on a wide Au(788) substrate and allowed to relax initially. Once the geometry attained (at least) a local minimum in energy, the system was evolved with a Langevin thermostat at 373.15K with a damping constant of 2.89 ps. The simulation was run for 2.5 ns, and the results are shown in Figure 5-8.
Figure 5-8: Chain formation at an Au(788) step site at 373K. (a) Shows the initial configuration of molecules, and (b) shows the definite alignment of some of the molecules along a step edge and chain formation. (c) Is an adapted STM image from [6] that shows the similarities with the results obtained here.

Figure 5-8 indicates that there is a definite preference for the molecules to use the step edges as an anchoring mechanism for chains to form. Also, the chains are relatively stable: while some of the chains deformed and the molecules diffused away, the upper-left chain in Figure 5-8(b) was stable, with only lateral movement of the chain. This simulation was ran
for a low coverage of TTPO; It is likely that higher coverages would show a more stable, organized chain substructure assembly like the ones shown in Figure 5-8(c). Still, these preliminary results show good agreement with these STM results that Au(788) does have potential to serve as a template for self-assembly of TTPO on gold.

5.3 Partial Monolayer Coverage on a Flat Surface

STM images in [6] show that one expects diffuse TTPO to be unstable on a flat, pristine surface, and there to be little to no organization in assembled substructures. Therefore, it warrants investigation with simulation to see the details of what occurs. A three-layer slab was constructed, with periodic boundary conditions in the x and y directions, and a fixed boundary in the z direction (to simulate vacuum, the conditions in which TTPO has been investigated previously [6]). Dynamics simulations were run in the NVT ensemble for temperatures $T=100K$, $200K$, $300K$, $400K$, $500K$, $600K$, $700K$ for 1.25 ns. After allowing the system to relax to a local minimum of potential energy (to minimize initial potential energy and avoid unrealistic dynamics or molecules having too much energy), the simulations were run with a timestep of 0.25 fs and a damping time constant of 12.493 picoseconds.

5.3.1 Cluster Formation

One thing that was prevalent among the pristine surface calculations at all temperatures were the formation of disorganized, amorphous, mobile clusters of molecules. This main substructure coexists with diffuse TTPO molecules (alone or in small chains of only a few molecules) that are far less stable. While the clusters tend to be amorphous and unstable, there still exists a degree of order within each substructure. There is a strong preference
for the planes of the molecules to align parallel to each other, such that chains are far more energetically preferred to the diffuse state. There were three main substructures observed within a cluster that the molecules assumed, an example of each is displayed in Figure 6:

(i) The energetically-preferred aligned formation (dubbed “chain formation” here, most stable).

(ii) A state in which the diffuse molecules adhere to the side of a chain perpendicularly (dubbed “side formation”).

(iii) A state in which the diffuse molecules attach themselves adjacent to the chain, forming a zipper-like pattern (“zipper formation”).
Figure 5-9: The three formations found in clusters of TTPO molecules on gold. Clusters can be found in any of the three states, or combinations of the three. The most stable state is the chain formation.

The chain formation is the desired state, since this state has the molecules aligned in the most organized way with their dipole moments aligned, and it is also the most stable. Instances of side formation were only metastable, with the molecules on the side navigating the side of the chain until they reach an end, then add themselves to the chain. Depending on the complexity of the cluster, the molecules on the side can also be contorted into the zipper formation. Molecules in a zipper formation undergo more complicated trajectories. The most observed scenarios for a cluster in zipper formation were the molecules oscillating between zipper and side formations, the molecules navigating themselves to the end of a chain, or there was chain swapping between two rows of the zipper. For example, if one
observes Figure 6(c), the very right-hand side of the image shows a chain of molecules that turns into a zipper formation near the center of the image. The chain can switch to the two sides of the zipper, since, when the chain is close to the middle of the two rows of the zipper, the energy difference becomes negligible and a switching can occur. It is important to note that the processes described here are thermally-dictated processes, and therefore are very sensitive to temperature. For example, in the lower (T=100K and 200K) temperature simulations, clusters that formed after the initial grouping maintained their size, location, and formation for the most part. Only lone and few-molecule chains (less than five or so) even changed location, since these are in an unfavored state. At higher T, even molecules in a chain can break off from the chain and join other groups of molecules, etc. So, the behavior described here is fickle and depends heavily on temperature.

While the processes that dictate cluster formation are complicated, one thing is certain: if there is hope for tailoring surfaces of TTPO molecules on gold, there needs to exist external methods of controlling the substructure formation. The pristine surface led to behavior that was unpredictable, unstable, disorganized, and unfavorable for self-assembled monolayers.

### 5.3.2 Activation Energy for Diffusion

MSD data was taken for several temperatures of a flat, wide, atomically smooth Au terrace (three layers) with 48 TTPO molecules placed on it in pseudo-random orientations (molecules were placed close to surface so that they would adsorb). The tilt angles and initial facing direction were arbitrary. The simulation was constructed by making a smaller slab (1/4 the size of the total slab) of gold with 12 molecules placed on top of it, then using the LAMMPS “replicate” command to double the simulation in the x and y directions. This has the effect of easily quadrupling the system size so that a higher number of molecules can
be used for statistical averaging. The timestep used was 0.2493 fs and the damping parameter for the NVT fix was 12.493 ps. The length of the simulation was different for some of the temperatures. While the lower temperatures were stable through 30,000,000 timesteps (7.5 ns), the higher temperatures (340K, 360K, 420K, 500K, 600K) grew unstable after a few million timesteps and atoms flew out of the simulation box. It is unclear why this was happening. Several different minimization styles were used to relax the system initially, but all led to unstable dynamics after a certain time. Since this only occurred at higher temperatures (340K and above), and never at low temperatures, it is likely that this is a thermal effect of relatively high temperature, and an effective evaporation was occurring (where the molecule received too much energy from the substrate and/or neighboring molecules and flew off of the surface). By default, when a molecule flies out of the simulation box (here, the z direction since it is a fixed boundary in that direction), the simulation stops. One can get around this, but it was decided that every simulation would have the same number of molecules present at the end of the simulation as at the start. Despite this, the trajectories persisted for several million timesteps at the appropriate temperature with MSD data that is in line with the more stable simulations. After initial energy minimization, the system was heated to a temperature $T$, and MSD data was outputted every 10,000 timesteps for the center of mass (COM) of each molecule, then the MSDs for each molecule were averaged together at each timestep to obtain the average MSD for that timestep.

As discussed before, the MSD vs. time for a system at constant temperature is expected to be linear. However, a plot taken from zero time and zero initial velocity will have two regimes. The first portion will be a steep increase in MSD from temperature equilibration. The plot will then approach a linear, or close to linear, portion that maintains a fairly constant slope. The best fit line is chosen as to exclude the portions of plots in which the
system is equilibrating such that the MSD data analysed is for the system at the desired temperature.

The data for a typical run is shown for 320K along with the associated best fit line and temperature in Figures 5-7 and 5-8.

![MSD, 320 K](image)

Figure 5-10: MSD data for the 7.5 ns simulation at 320K. The green dots are the data used for diffusion coefficients ($dx^2 + dy^2 + dz^2$), the red circles are the $dx^2$, and the blue x’s are $dy^2$.

For each MSD plot, care must be taken when choosing which portion of the plot to place the linear fit. It should be after the system is properly at the correct temperature, and also after it has been at that temperature for an appropriate amount of time such that the system has reached a steady-state diffusion at that temperature. Usually, there will be an obvious linear portion in the MSD plot at long times that is the one to be chosen.
Figure 5-11: (a) Temperature vs. time for the run for comparison to temperature equilibration regime. (b) MSD with the best fit line shown. The excluded thermalization data are in red.

The data for the linear fits and the corresponding diffusion coefficients are detailed in Table 5.1. **Note:** Since surface diffusion occurs primarily in the x-y plane ($dz^2 \approx 0$), it is a better approximation to set the MSD slope proportional to $4Dt$, where D is the diffusion coefficient instead of $6Dt$ (in three dimensional diffusion).
Table 5.1: Diffusion Coefficients for TTPO on Flat Surface

<table>
<thead>
<tr>
<th>T(K)</th>
<th>Slope (Å$^2$/fs)</th>
<th>y-intercept (Å$^2$)</th>
<th>Diffusion Coefficient (cm$^2$/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>1.198 × 10$^{-5}$</td>
<td>282.3</td>
<td>2.995 × 10$^{-7}$</td>
</tr>
<tr>
<td>260</td>
<td>2.07 × 10$^{-5}$</td>
<td>261.4</td>
<td>5.175 × 10$^{-7}$</td>
</tr>
<tr>
<td>280</td>
<td>4.815 × 10$^{-5}$</td>
<td>330.6</td>
<td>1.204 × 10$^{-6}$</td>
</tr>
<tr>
<td>320</td>
<td>6.064 × 10$^{-5}$</td>
<td>370.2</td>
<td>1.516 × 10$^{-6}$</td>
</tr>
<tr>
<td>340</td>
<td>0.000359</td>
<td>264.6</td>
<td>8.975 × 10$^{-6}$</td>
</tr>
<tr>
<td>360</td>
<td>0.000248</td>
<td>207.4</td>
<td>6.2 × 10$^{-6}$</td>
</tr>
<tr>
<td>420</td>
<td>0.000622</td>
<td>-253.2</td>
<td>1.555 × 10$^{-5}$</td>
</tr>
<tr>
<td>500</td>
<td>0.00037</td>
<td>311.9</td>
<td>9.25 × 10$^{-6}$</td>
</tr>
<tr>
<td>600</td>
<td>0.0033</td>
<td>-1287</td>
<td>8.25 × 10$^{-5}$</td>
</tr>
</tbody>
</table>

Following the strategy of Section 4.2.2, the natural logarithm of the diffusion coefficients can be plotted against $1/k_B T$ to yield the activation energy associated with TTPO diffusion on a flat gold surface. This Arrhenius Plot is shown in Figure 5-9.

![Arrhenius Plot, Flat Terrace](image)

Figure 5-12: Arrhenius plot for a flat, smooth, three-layer gold slab. The parameters for the fit (with 95% confidence bounds in parentheses) are: slope = -0.142 (-0.1919, -0.092) eV, y-intercept = -7.462 (-9.304, -5.619).

From comparison of the parameters given in Figure 5-9 and Equation 4.17, the activation energy for TTPO diffusion is 0.142 eV, and the pre-exponential factor is

$$e^{-7.462} = 0.000574 \text{ cm}^2/\text{s} = D_0$$
Arguably the most useful quantity of the Arrhenius plot is the activation energy. The pre-exponential factor is of relatively little interest. Essentially, it is the diffusion coefficient for a TTPO molecule when the exponential is equal to 1, meaning that $E_a=0$ or $T \to \infty$. Since $E_a=0$ is impractical, it is more useful to interpret it as the diffusion coefficient for a molecule of TTPO at very high temperatures. Since the thermal breakdown temperature of TTPO is 450° (723.1 K), the quantity $D_0$ provides little insight into the energetics of TTPO surface diffusion.

However, $E_a$ is a more interesting quantity here. First, the value 0.142 eV is about an order of magnitude higher than that reported for pentacene in [75], where it was found that pure pentacene faces a diffusion barrier of 0.041 eV in the x and y directions (calculated from DFT-based simulations). The value obtained here is consistent with this data, since the S-Au interaction in TTPO is much stronger than the interaction of the Au surface with pure pentacene, so TTPO requires more energy to essentially abandon its present site and jump to a new site.

By inspection of the data in Figure 5-9, it is clear that the mechanisms by which TTPO moves across a gold surface is not trivial. For a simple monatomic, non-interacting (with other adatoms) particle on a surface, the data would better trace out a straight line on an Arrhenius plot with little deviation. The activation energy would then likely correspond to a hopping from a local minimum in an interstitial site to an adjacent site, or a similar simple diffusion scheme. However, TTPO is a relatively large molecule with a complicated interaction profile with the substrate and with other adsorbed molecules. While the plot doesn’t provide the exact methods in which TTPO moves, it, at least, provides an estimate for the energies involved with TTPO diffusion. For example, compared with room temperature ($\approx 0.025$ eV), the activation energy calculated here is about an order of magnitude higher.
However, TTPO has been seen via STM to diffuse at room temperature [6]. Therefore, other factors are surely involved. This is likely a combination of the substrate supplying energy to a molecule (so that it has sufficient energy to jump sites), and lateral effects from the substrate and other molecules (a “pulling” from the other molecules and laterally-positioned substrate atoms). Hence, while surely affected greatly by thermal effects, it is not a purely thermally-activated diffusion.

5.4 Site Vacancies

As substrate vacancies (single or multiple missing atoms) represent a break in the surface energy periodicity, it is worth examining the effect of these defects on the assembled TTPO structures. For this, the same unit cell as above was used (three gold layers with 48 TTPO molecules randomly placed on top in same random initial configurations). However, an atom was removed from the top layer near the middle of the cell (before replication in x and y directions), such that after replication there were 4 vacancy sites, each in the middle of each “quadrant” of the total slab, see Figure 5-10.
The simulations were then run under identical conditions as detailed above for 2.5 ns at 200K and 300K in the NVT ensemble, and 500K and 600K with Langevin Dynamics (since molecules in the NVT desorbed and flew out of the simulation box, which triggers LAMMPS to stop). Damping constants of 3.11 ps and 23.1 ps, respectively, were used. These were derived from Equation ?? using the molecular mass of one TTPO molecule, \(2.32 \times 10^{-23}\) kg, with the diffusion coefficients in Table 5.1. Analysing the trajectories showed that there was little, if any, interaction with the vacancies at all sampled temperatures. While there was some evidence that one or more molecules may have been at least partially anchored to a vacancy, there was no evidence that a single dislocation by itself was sufficient to permanently anchor a molecule. If a TTPO’s trajectory was even possibly altered by a hole, the halting was not stable and the molecule eventually continued to diffuse seemingly unaltered. For surface vacancy defects to be considered as a tailorable means of designing surface architectures, the anchoring should have an overwhelming and obvious effect on the molecules’ and clusters’ trajectories. As this was not the case, further investigation is needed.
for different vacancy patterns.

5.4.1 Substrate Trenches: The Lock-and-Key Effect

While a single surface vacancy was not seen to have any significant impact on the mobility of a single TTPO molecule or a cluster, an interesting effect happened when a line of six adjacent missing atoms were created (imagine a red circle to the left and right of each red circle in Figure 5-13, so that effective trenches six atoms and 20.1Å across were created as opposed to a single hole). The TTPO molecule is approximately 13.6Å across, so the trenches are large enough to accommodate a molecule but not too large such that the molecule can displace very much laterally within it. The simulations were carried out in the same setup as above: four simulations at the same four temperatures, only this time the higher temperatures were run in the NVT ensemble as well. A molecule desorbed in the 500K simulation, but it was late in the simulation (1.6 ns out of a 2.5 ns simulation), so dynamics were stable for a long enough time so that phenomena were witnessed.

For the lower two temperatures, no events of interactions with the trenches were observed. This is likely due to the relatively low diffusivity; The molecules simply don’t spend very much time near the trenches and stay relatively localized to their locations after thermalization. Molecules in the 300K simulation did get near the trenches in some instances, but showed no apparent reaction to them and there was no evidence that the trenches had any effect on their trajectories.

The higher two temperatures, however, witnessed an interesting event. There was an instance in the 500K simulation and in the 600K simulation of a molecule being adjacent (to the side of the horizontal trench) and parallel to the trench, then being absorbed into the hole. Once here, the molecules remained there for the entirety of the simulation, and neighboring
molecules were attracted to this anchored molecule. What is peculiar is that in the 300K, 500K, and 600K simulations, molecules that approached a trench either perpendicular or not completely parallel with the trench were visually unaffected by it, leading one to deduce that the absorption of a molecule into the trench obeys a “lock and key” mechanism, where the alignment of a molecule on the surface must match the orientation of the trench exactly or near exactly. See Figure 5-14
Figure 5-14: Three snapshots in time showing the stability of a molecule in a trench site. In all cases, the molecule that reacted with the trench is bordered with a red rectangle. (a) Shows the instant before the molecule becomes absorbed or “falls” into the trench. Note also that the molecules near the lower left trench approach it but do not appear to react to it, despite having several molecules near it. (b) Shows the molecule right after absorption into the trench. (c) Shows the system at a much later time (over 1 ns later, a relatively long time for simulations), and the molecule is completely stable in the trench. A long cluster seems to form off the anchored molecule, although one can see that it is not very organized and it was mobile.

A similar phenomenon has been seen before for the organic molecule Violet Lander (C_{108}H_{104}) [76]. In the combined STM/ molecular dynamics study, it was found that by
simply altering the orientation of the molecule with respect to the substrate, the diffusion coefficient was reduced by two orders of magnitude. While the study did not focus on vacancy sites, it still shows that the idea of a molecule interacting with the surface can happen in some circumstances only for certain orientations of the molecule. The idea is similar to the lock-and-key effect seen with enzymes: an enzyme will only react with a substrate when the substrate (key) exactly fits the enzyme (lock). Here, the molecules (key) only seemed to react with the gold substrate trench (lock) when the orientation of the molecule was parallel and directly next to the trench. Otherwise, it seemed to be unaffected.
Chapter 6

Conclusions

6.1 Overview and Summary

This project investigated some of the underlying science and mechanisms of directed self-assembly with applications to photovoltaic devices through molecular dynamics simulations in two very different frameworks. The work here supplements earlier experimental and computational work [6] on the self-assembly of 5,6,7-trithiapentacene-13-one on flat, wide Au(111) terraces and vicinal Au(788) step edges. By adding this new molecular dynamics simulations take, there is a more complete picture of not only the substructures TTPO forms on these surfaces, but the exact ways in which it forms them.

The project started with first understanding how best to implement these systems in both a quantum-based framework (in QE) and a classical mechanics framework (in LAMMPS). Calculations in QE are first-principles based, so the full electronic picture is painted. However, this level of detail came at a cost, and the calculations were very slow and only accommodated a relatively small number of atoms. Still, there was useful information to be extracted from them. First, the tilt angle of 11.5° and gold-sulfur distance of 2.5Å on four layers of stationary gold gave good agreement with the value of 14.1° and 2.3Å found from
classical simulation. Also, analysing qualitatively a small, 5 ps dynamics run at 300K side-by-side with an identical unit cell in LAMMPS showed the same behavior on this timescale, with the molecules reaching the same initial tilt and following the same pattern of staying localized and oscillating around a small angle afterwards. Finally, the first-principles calculations gave the first insight into a preferential direction of alignment and travel for the TTPO molecules. The molecule clearly aligned its sulfur bridge along a direction of the sixfold symmetry of the flat gold substrate and then moved along that direction, indicating that the substrate does indeed have a significant effect on the mobility of the molecules.

Although QE calculations were limited by their inherent inefficiency, they still instilled confidence in the good parameterization of the LAMMPS model, which was a bigger gateway into being able to understand large-scale behavior of a system of TTPO molecules. Before being able to study these various systems, however, the force field and molecular model had to be constructed. By finding the appropriate parameters for the corresponding interactions in the various chemical environments from the AMBER94 and GAFF force field databases, the TTPO molecule was constructed through a total of 42 bonds, 58 angles, and 101 dihedrals. This, combined with all-atom (not course-grained, all atoms were treated as their own degrees of freedom) simulations ensured that the molecule would have full flexibility and be described with a high amount of detail.

After thorough construction of the TTPO molecule itself, a force field was developed with the parameters found in Table 3.4. This was done with thorough research within the AMBER databases and appropriate external literature. Long-range interactions were described through a combined 12-6 Lennard Jones potential and a Coulombic potential with the partial charges shown in Figure 3-5. Pairwise interactions were described by an asymmetric Morse potential for Au-C, Au-S, and Au-H, and by LJ through appropriate
mixing rules otherwise. The gold atoms were treated by the very successful Embedded Atom Model, in which LAMMPS distributes an appropriate potential file for gold.

Following the proper construction of the TTPO molecule and its corresponding interaction force field with the gold surface and other TTPO molecules, large-scale phenomena of partial monolayer coverages were studied. The NVT ensemble was used to thermostat systems of temperatures up to about 340K, and a Langevin thermostat described the higher temperature systems in which molecules flew off the surface and out of the simulation box. First, diffusion coefficients were calculated for several temperatures of 48 TTPO molecules on a flat Au terrace, and are shown in Table 5.1. By plotting the diffusion coefficients verses inverse temperature, an activation energy for diffusion of 0.142 eV was found for TTPO on flat gold, which is an order of magnitude higher than that reported for pentacene in [75]. This is consistent due to the much higher binding energy of the sulfur bridge with the gold surface than the (relatively weak) pure pentacene interaction with the surface. In addition to analysis of the energetics, the phenomenon of stable chain formation at the Au(788) edges were directly witnessed via simulation. The simulated chains were in excellent agreement with the DFT simulations and STM images in [6], with clear offset structure and chain stability.

Finally, new phenomena previously unstudied were analysed for a system of TTPO molecules. A point vacancy was created at four spots in a large gold slab, and a partial monolayer of TTPO was placed on the slab in random orientations. After being thermalized at 200K, 300K, 500K, and 600K, the point defect did not appear to have any obvious effect on the trajectories of the molecules. In order to be a suitable candidate for anchoring molecules to become a basis for self-assembled chains to form, the site defects should have kept a molecule stationary. However, this was not seen at any temperature. However, when
trenches that were 20Å across were created by removing six adjacent atoms at the same four locations, a profound effect was seen at 500K and 600K in which a molecule was absorbed down into the trench and held there for the duration of the experiment. Other molecules were attracted to the stationary one and formed disorganized, relatively stationary clusters around it, in contrast to highly mobile and amorphous clusters without this trench basis. So, it can be concluded that these trenches could be a means of immobilizing a TTPO molecule in order to perhaps begin chain formation.

### 6.2 Possible Future Work

There are a few directions that follow-up simulation work to this project could go:

1: More finite-temperature DFT calculations of several molecules on several layers of gold could provide a more complete comparison with classical results. While the results obtained in this study were insightful, due to the time constraint of the project, only small unit cells were studied in order to see small, short-term diffusive events. For example, six to eight molecules placed on a step edge so that they span two terraces simulated for many (perhaps 50-100) picoseconds may yield a first-principles confirmation of the stable assembly of gold. This calculation would likely take a long time (months), but it would be yet another confirmation of the steps serving as a template for stable assembly.

2: Charge density calculations for molecules on the various substrates might give insight into the diffusive and assembly nature of the TTPO molecules. For instance, why are molecules at the step edge and at the corner of a step stable while those on the wide terraces are so mobile? It is likely due to the break in symmetry provided by the steps, but a mapping of the electronic structure of these states might be insightful into helping to answer, more
specifically, why the structures occur.

3: Further analysis of the effects of the substrate trenches reported in Chapter 5. The defects were very successful in immobilizing the molecules they trapped, it is worth investigating further the possible ability to form stable nanowires that could potentially begin at one trench site and extend to another, or possibly a step edge. Also, to quantify its ability to hinder the diffusion of a system of TTPO molecules, diffusion coefficients could again be calculated at various temperatures to see if the trenches have a large impact on molecular mobility.

4: Higher monolayer coverages and greater than one monolayer coverages simulations could possibly lead to results that are in agreement with the STM results reported in [6]. Also, more than coverages of more than one monolayer will start to show how stable TTPO is in its head-to-tail-stacked crystalline form, which is desired for high-temperature photovoltaic applications. Higher-layered coverages will likely be most stable on Au(788) steps given the current data, but it would be interesting to see how multiple layers of TTPO behave on flat gold as well.
APPENDICES
Preparing a data file that can be easily manipulated and interpreted by LAMMPS is a multi-step and multi-software process. The softwares used are VMD, TopoTools (included with latest versions of VMD), and Moltemplate. While there are several ways that this can be achieved, the following outlines how data file generation and tailoring was performed in this project, with the steps being in order of how they were performed. It should be noted that this procedure purely edits the topology information for a calculation, and the parameterization of force fields, bonds, dihedrals, etc. is a separate matter.

0: An .xyz file with atomic species and coordinates was manually prepared. The atomic coordinates for TTPO were supplied to me by a past researcher who successfully modelled TTPO via DFT methods [6]. A small slab of gold was prepared manually in Excel with the same lattice constant as [6] as to maintain consistency. This is listed as Step 0 because the base atomic coordinates and structure information was entered manually, as to be consistent with both past work and physical data. This Appendix outlines how to manipulate topology information for a LAMMPS data file, but knowledge of the structure is required beforehand, and here it was prepared in an .xyz structure file with atomic coordinates used in both the DFT calculations in this work and that in [6] and the present work in LAMMPS. There is an .xyz file for TTPO (“TTPO.xyz”) and one for a small gold slab (“Au.xyz”).

1: After an .xyz file of each component of a system was made, it can be uploaded into VMD. VMD has the ability to visualize many file types, and the structure was visible in the display window. After checking that it looked correct, the Tk console took was opened via “Extensions→Tk Console” in the “VMD Main” window. Although TopoTools should come pre-installed with the latest editions of VMD, its installation is verified through the command “package require topotools” in the Tk console. This should return a number, which is the version of TopoTools installed. After verifying this TopoTools is installed, a LAMMPS data file (“TTPO.data”) (not necessarily the final data file to be used in a calculation) is generated using the TopoTools command

topo writelammpsdata TTPO.data

This reads the .xyz file for the TTPO molecule, then writes its (purely structural) coordinate data to a LAMMPS-style data file.

2: At this point, the data file will have three sections. The first is the header, which
contains information about the number of atoms (36 for the TTPO molecule), number of bonds, angles, dihedrals, and impropers (all 0 at this point), atom types (5), and bond, angle, dihedral, and improper types (all 0 at this point), and contains the unit cell boundaries (xlo xhi ylo yhi zlo zhi). The second sections is the Masses section, which contains the mass of each atomic species (amu), and defines the atomic type (a number, 1-5 for TTPO, used as a label). The final section is the Atoms section, which has a row for each atom in the TTPO.xyz file. The line is in the format:

(Atom Number) (Molecule Number) (Atomic Type) Q X Y Z

Where Q is the charge, and X,Y,Z are the atomic coordinates.

3. The dihedrals, angles, and bonds must now be inputted manually in the corresponding sections in the corresponding section in the data file. While this step may not be necessary for every molecule (it is not necessary when making a gold slab, for example), it is needed for the TTPO molecule specifically. For the TTPO molecules in this study, these parameters were constructed from the geometry of the TTPO molecule by analysing its structure. Each molecule contained 42 bonds (7 bond types), 58 angles (4 angle types), and 101 dihedrals (5 dihedral types). Table 3.4 shows the different types of each of the parameters and the corresponding values of the potential energy terms. While there are ways for software like TopoTools to guess these interactions, it can often be unreliable without being well-versed in the software, so it was found to be easier to manually construct these interactions and input them directly into the data file. After it was done once, the molecule was copied all times in the future.

Note: If a molecule is to be simulated by itself with no other contents in the unit cell, then this data file is ready to be used in LAMMPS (after setting the cell boundaries). However, often multiple pieces are required in a unit cell so further construction is needed.

4. Moltemplate contains a utility called ltemplify.py that is a Python script used to convert LAMMPS data files into .lt files (those used by Moltemplate). The function is used via the command line in the /src subdirectory within the main Moltemplate directory. To create a file TTPO.lt from TTPO.data, one uses the command

ltemplify.py -name TTPO TTPO.data > TTPO.lt

The file TTPO.lt defines a TTPO molecule. Within it are information about charges, atomic coordinates, dihedrals, angles, and bonds. This can now be inserted and easily manipulated as part of a system.

5. One now creates a file called “System.lt”. This will be the file that puts together all the pieces of the unit cell (the data file that will be used in a LAMMPS simulation). On the top line, one imports the TTPO.lt file via the command “import TTPO.lt”. To actually place a TTPO molecule in the cell, use the command
TTPO1 = new TTPO

and so on for however many TTPO molecules are desired in the simulation. **Note:** If one wishes to also simulate a gold slab, “import slab.lt” would be issued if a slab.lt file was created following the preceding steps. From here, each molecule can be translated and rotated by appending a .move and .rot command:

TTPO1 = new TTPO.move(x,y,z).rot(θ,x₀,y₀,z₀)

which first moves the molecule by x,y,z Angstroms then rotates it θ degrees by an axis that passes through the origin and the point (x₀,y₀,z₀). This is done for each molecule until the desired geometry is reached.

6. The final step is to now convert the System.lt file into a data file to be used by LAMMPS and visualize it in VMD. This is done through the command

moltemplate.sh -atomstyle “full” -nocheck -vmd System.lt

This command does three things. 1. It sets the atom style to full, which is a LAMMPS atom style that incorporates charge. 2. It creates a file System.data which can now be used in a LAMMPS simulation (after setting the boundaries) and represents a system of multiple components. 3. It opens the system in VMD for visualization. It is crucial to view each system before every simulation to ensure that the geometry is that desired.
APPENDIX B
EXAMPLE LAMMPS INPUT FILE

The following is an example of an input file in LAMMPS. This specific calculation calls the
data file “monof.data”, which is a monolayer of TTPO molecules on a three-layer gold sub-
strate. A single or many #’s indicate a comment. Sections of this input file are separated by
lines of # symbols. To summarize: The first section defines the simulation setup (interaction
styles, units, boundary conditions, dimension, etc.), the second only reads in the data file
(or a restart file if starting from a previous run is desired), the third defines all the force
field parameters and defines groups and fixes that keep atoms stationary (non-surface gold
atoms), and the last couple sections define simulation-specific parameters (time steps, dump
files, computes, minimizations, run time, etc.). A couple of notes: while LAMMPS allows
one to set force field parameters in the data file, this was never done. The parameters were
set in the input file and not altered. To manipulate a simulation with the same physical
system then only the last two sections should be altered, since those are the ones that define
simulation-specific inputs.

*******************************************************************************
### General Simulation Setup
*******************************************************************************
dimension 3
boundary p p f
units metal
atom_style full
pair_style hybrid lj/cut/coul/cut 10 lj/cut 10.0 morse 5 eam
pair_modify pair lj/cut mix arithmetic
bond_style harmonic
angle_style harmonic
dihedral_style harmonic

*******************************************************************************
### Read in LAMMPS data file
### Atomic coordinates and bond/angle/dihedral defined
read_data monof.data
*******************************************************************************
### Setting Force Field Parameters
### Atom type and symbol: 1-Au 2-C 3-H 4-O 5-S
### Gold-Gold interaction
pair_coeff 1 1 eam /home/spg/rmiller/lammps_source/potentials/Au_u3.eam

### Gold-Sulfur interaction
pair_coeff 1 5 morse 0.138 1.38 2.903

### Gold-C
pair_coeff 1 2 morse 0.0096 1.0137 4.104

### Gold-H
pair_coeff 1 3 morse 0.0031 1.166 4.006

### Gold-O Van der Waals
pair_coeff 1 4 lj/cut 0.00392 2.946

### i,i type pairwise VdW interactions
pair_coeff 2 2 lj/cut/coul/cut 0.00372 3.399
pair_coeff 3 3 lj/cut/coul/cut 0.00065 2.599
pair_coeff 4 4 lj/cut/coul/cut 0.00910 2.959
pair_coeff 5 5 lj/cut/coul/cut 0.01084 3.563

bond_coeff 1 24.72 1.229 #C=O
bond_coeff 2 12.5799 1.55 #C-C
bond_coeff 3 15.914 1.08 #C-H
bond_coeff 4 10.563 1.79 #C=S
bond_coeff 5 7.1980 2.038 #S-S
bond_coeff 6 20.340 1.400 #C=C
bond_coeff 7 14.262 1.675 #C=S

angle_coeff 1 1.51 120 #h-c-c
angle_coeff 2 2.73 120 #c-c-c
angle_coeff 3 2.69 120 #s-c-c
angle_coeff 4 3.11 120 #o-c-c

dihedral_coeff 1 0.157 -1 2 #h-c-c-c
dihedral_coeff 2 0.157 -1 2 #h-c-c-h
dihedral_coeff 3 0.157 -1 2 #c-c-c-c
dihedral_coeff 4 0.130 -1 2 #o-c-c-c
dihedral_coeff 5 0.157 -1 2 #s-c-c-c

thermo 100
thermo_style one

group TTP0 type 2 3 4 5
group gold type 1
group bottom1 id 433:504
group bottom2 id 577:648
group bottom3 id 721:792
group bottom4 id 865:936
group bottom5 id 1009:1080
group bottom6 id 1153:1224

fix freeze1 bottom1 setforce 0 0 0
fix freeze2 bottom2 setforce 0 0 0
fix freeze3 bottom3 setforce 0 0 0
fix freeze4 bottom4 setforce 0 0 0
fix freeze5 bottom5 setforce 0 0 0
fix freeze6 bottom6 setforce 0 0 0

group mobile subtract all bottom1 bottom2 bottom3 bottom4 bottom5 bottom6

group defect id 99
delete_atoms group defect
replicate 2 2 1

###############################################################################
### Running Simulation
###
#
min_style sd
minimize 1.0e-20 1.0e-20 100000 100000
#
# Thermostat/ Set computes
set group gold mol 0
compute chunk TTPO chunk/atom molecule discard yes
compute chunkmsd TTPO msd/chunk chunk
fix chunkmsd TTPO ave/time 1 1 10000 c_chunkmsd file monofd.500.msd mode vector
compute temp1 mobile temp
fix 8 mobile nve
fix 9 mobile langevin 500 500 12.493 12345
thermo_modify temp temp1
fix_modify 9 temp temp1
compute rdf TTPO rdf 100
fix rdf TTPO ave/time 1 1 10000 c_rdf file monofd.500.rdf
mode vector
timestep 0.00024987
#
# Dump files and run
dump dump all atom 4000 monofd.500.lammpstrj
run 5000000
write_restart monofd.500.re
Installing Quantum ESPRESSO version 5.1.2 was done following precisely the same process outlined in Appendix K in [6]. Instructions for how LAMMPS was built for the parallel environment on Trillian is detailed here. First, LAMMPS was installed in the home directory on Trillian:

/mnt/lustre/lus0/home/spg/rmiller/

LAMMPS must be built with either just the default packages (listed in [41]), or with additional optional packages with the command

make yes-(optional package name)

In order to build an executable for LAMMPS, one has to edit or construct a proper Makefile. There are several distributed with LAMMPS for both serial and parallel computing, and some are even pre-made for specific machines. However, building the executable used for parallel execution on LAMMPS required the following Makefile (for the executable mine2):

```plaintext
# mine = default MPI compiler, default MPI

SHELL = /bin/sh

# Path to src files

vpath %.cpp /mnt/lustre/lus0/home/spg/rmiller/lammps_source/src
vpath %.h /mnt/lustre/lus0/home/spg/rmiller/lammps_source/src

# ------------------------------
# compiler/linker settings
# specify flags and libraries needed for your compiler

CC = CC
```
CCFLAGS = -fastsse
SHFLAGS = -fPIC
DEPFLAGS = -M

LINK = CC
LINKFLAGS =
LIB =
SIZE = size

ARCHIVE = ar
ARFLAGS = -rc
SHLIBFLAGS = -shared

# LAMMPS-specific settings, all OPTIONAL
# specify settings for LAMMPS features you will use
# if you change any -D setting, do full re-compile after "make clean"

# LAMMPS ifdef settings
# see possible settings in Section 2.2 (step 4) of manual

LMP_INC = -DLAMMPS_GZIP

# MPI library
# see discussion in Section 2.2 (step 5) of manual
# MPI wrapper compiler/linker can provide this info
# can point to dummy MPI library in src/STUBS as in Makefile.serial
# use -D MPICH and OMP settings in INC to avoid C++ lib conflicts
# INC = path for mpi.h, MPI compiler settings
# PATH = path for MPI library
# LIB = name of MPI library

MPI_INC =
MPI_PATH =
MPI_LIB =

# FFT library
# see discussion in Section 2.2 (step 6) of manual
# can be left blank to use provided KISS FFT library
# INC = -DFFT setting, e.g. -DFFT_FFTW, FFT compiler settings
# PATH = path for FFT library
# LIB = name of FFT library

FFT_INC = -DFFT_FFTW3
FFT_PATH = -L$(FFTWDIR)/lib
FFT_LIB = -lfftw3

# JPEG and/or PNG library
# see discussion in Section 2.2 (step 7) of manual
# only needed if -DLAMMPS_JPEG or -DLAMMPS_PNG listed with LMP_INC
# INC = path(s) for jpeglib.h and/or png.h
# PATH = path(s) for JPEG library and/or PNG library
# LIB = name(s) of JPEG library and/or PNG library

JPG_INC =
JPG_PATH =
JPG_LIB =

# ---------------------------------------------------------------------
# build rules and dependencies
# do not edit this section

include Makefile.package.settings
include Makefile.package

EXTRA_INC = $(LMP_INC) $(PKG_INC) $(MPI_INC) $(FFT_INC) $(JPG_INC) $(PKG_SYSINC)
EXTRA_PATH = $(PKG_PATH) $(MPI_PATH) $(FFT_PATH) $(JPG_PATH) $(PKG_SYSPATH)
EXTRA_LIB = $(PKG_LIB) $(MPI_LIB) $(FFT_LIB) $(JPG_LIB) $(PKG_SYSLIB)
EXTRA_CPP_DEPENDS = $(PKG_CPP_DEPENDS)
EXTRA_LINK_DEPENDS = $(PKG_LINK_DEPENDS)

# Path to src files
vpath %.cpp ..
vpath %.h ..

# Link target
$(EXE): $(OBJ) $(EXTRA_LINK_DEPENDS)
$(LINK) $(LINKFLAGS) $(EXTRA_PATH) $(OBJ) $(EXTRA_LIB) $(LIB) -o $(EXE)
$(SIZE) $(EXE)

# Library targets
lib: $(OBJ) $(EXTRA_LINK_DEPENDS)
$(ARCHIVE) $(ARFLAGS) $(EXE) $(OBJ)

shlib: $(OBJ) $(EXTRA_LINK_DEPENDS)
$(CC) $(CCFLAGS) $(SHFLAGS) $(SHLIBFLAGS) $(EXTRA_PATH) -o $(EXE) $(OBJ) $(EXTRA_LIB) $(LIB)
# Compilation rules

%.o: %.cpp $(EXTRA_CPP_DEPENDS)
$(CC) $(CCFLAGS) $(SHFLAGS) $(EXTRA_INC) -c $<

%.d: %.cpp $(EXTRA_CPP_DEPENDS)
$(CC) $(CCFLAGS) $(EXTRA_INC) $(DEPFLAGS) $< > $@

%.o: %.cu $(EXTRA_CPP_DEPENDS)
$(CC) $(CCFLAGS) $(SHFLAGS) $(EXTRA_INC) -c $<

# Individual dependencies

DEPENDS = $(OBJ:.o=.d)
sinclude $(DEPENDS)

There are a few key notes about this makefile. First, it was created for compilation with PGI. This is accessed as a module file in Trillian with the command "module switch PrgEnv-cray PrgEnv-pgi". The user is now in a PGI programming environment. Next, the top section includes paths to the source files (.cpp and .h) that LAMMPS looks for when building the executable. This is followed by compiler settings, linkers, MPI section (left blank, since the PGI module was loaded), and FFT library settings. Once this Makefile was created, LAMMPS was built with the command

make mine2

LAMMPS automatically finds the file Makefile.mine2 (if it is in the /src/MAKE directory) and begins to make the executable. If the process exits with no errors, then LAMMPS is properly built. The executable is now pointed to in a batch submission file and LAMMPS is ready to be used for parallel execution.
Bibliography


[56] TopoTools Main Page. [https://sites.google.com/site/akohlmey/software/topotools](https://sites.google.com/site/akohlmey/software/topotools).


