

# Introduction to Molecular Dynamics and Density Functional Theory

Alathea Davies, Dept. of Chemistry  
Summer 2024 – Bio-Comp Hackathon  
June 12, 2024



# Outcomes

After attending and participating in this lecture, students will be able to...

- ... list the basic steps of a molecular dynamics simulation.
- ... demonstrate the best practices for finding an appropriately suited force field for a specific system.
- ... identify limitations of molecular dynamics.
- ... execute and analyze the results of a basic molecular dynamics simulation of a solvated protein using GROMACS software.
- ... identify limitations of density functional theory.



# Molecular Dynamics



06/12/2024

Bio-Comp Hackathon 2024

2

UNIVERSITY OF WYOMING

# What is molecular dynamics?

Molecular dynamics (MD) is a simulation technique for computing the transport properties of a **classical**, many-body particle system.

A “classical” system means that the physics is deterministic. We use Newton’s equations of motion to calculate the properties of our system.

In contrast, a “quantum” system is probabilistic. Calculations of this nature, which attempt to solve the Schrödinger Equation, are known as first-principles or *ab initio*.

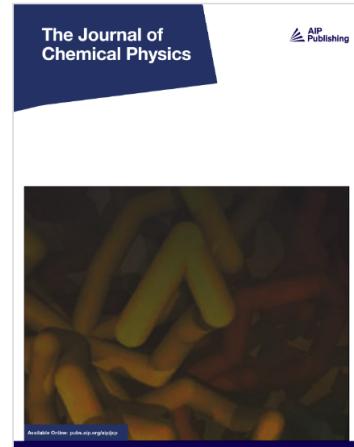


# What is molecular dynamics?

The first MD simulation was done by Alder and Wainright in the late 1950s'.

**Volume 31, Issue 2**

August 1959



RESEARCH ARTICLE | AUGUST 01 1959

## Studies in Molecular Dynamics. I. General Method

B. J. Alder; T. E. Wainwright



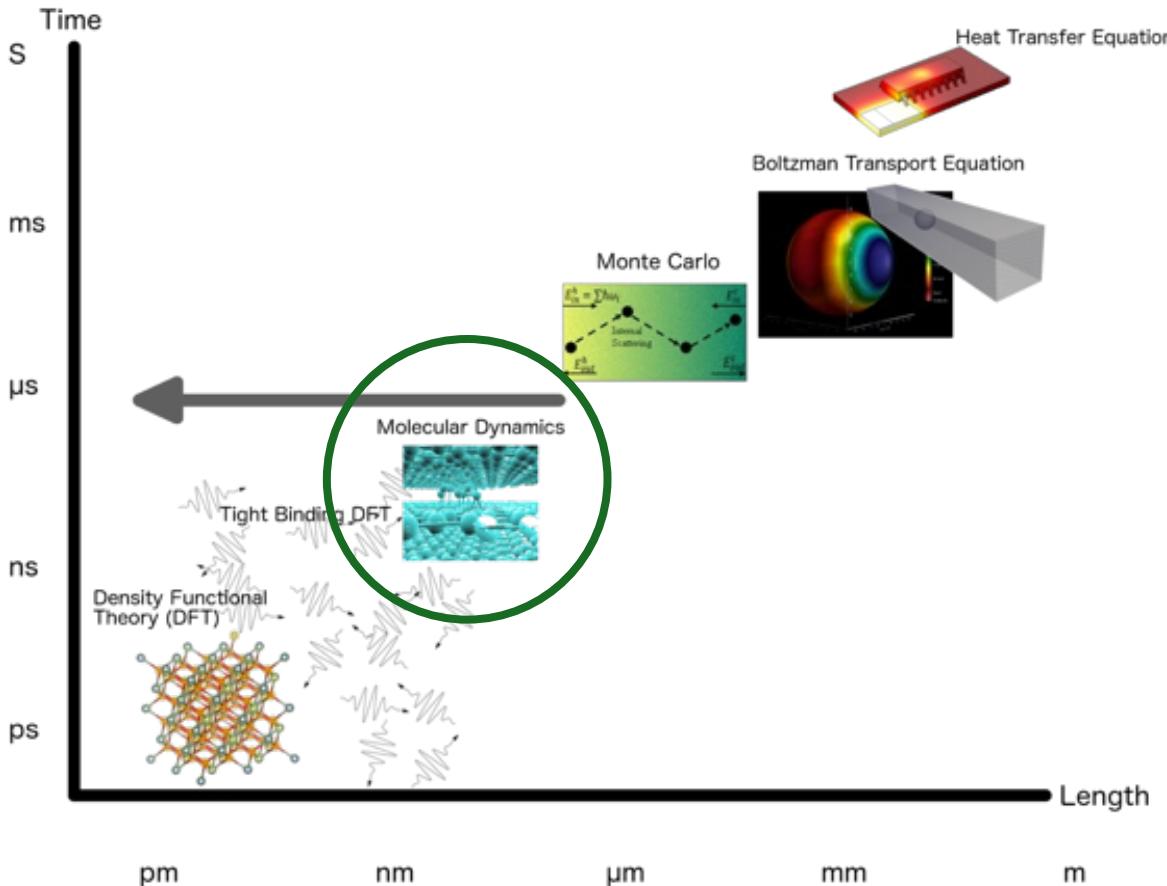
*J. Chem. Phys.* 31, 459–466 (1959)

<https://doi.org/10.1063/1.1730376> Article history 



A method is outlined by which it is possible to calculate exactly the behavior of several hundred interacting classical particles. The study of this many-body problem is carried out by an electronic computer which solves numerically the simultaneous equations of motion. The limitations of this numerical scheme are enumerated and the important steps in making the program efficient on the computers are indicated. The applicability of this method to the solution of many problems in both equilibrium and nonequilibrium statistical mechanics is discussed.

# When do we use MD?



MD allows us to look at larger systems and timescales than *ab initio* methods

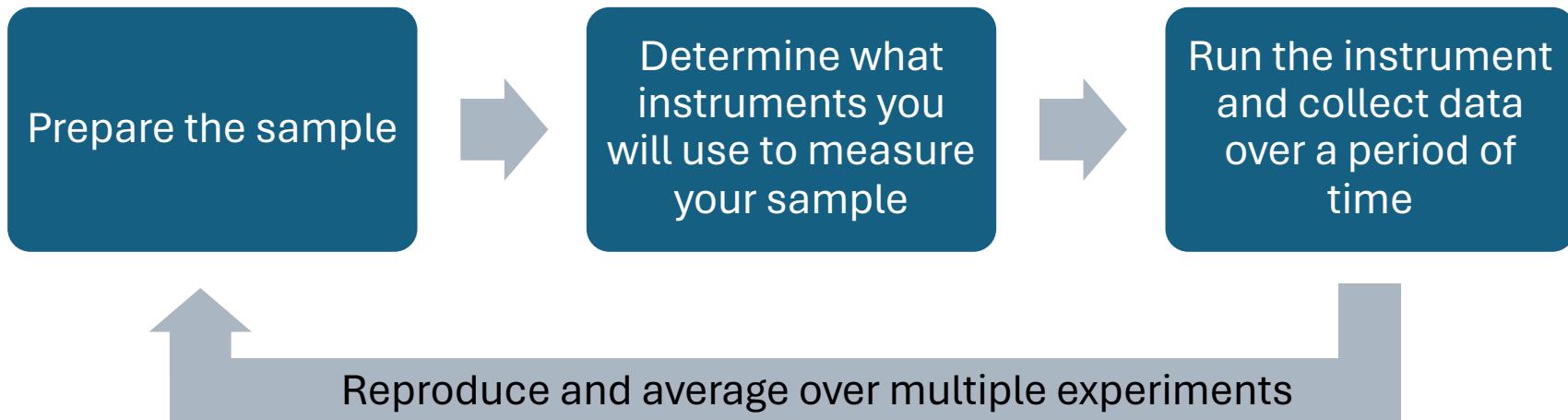
Although MD can be performed with *ab initio* methods (AIMD), this lecture will focus on classical MD.



# What are the steps of MD?

Running an MD simulation is like conducting an experiment.

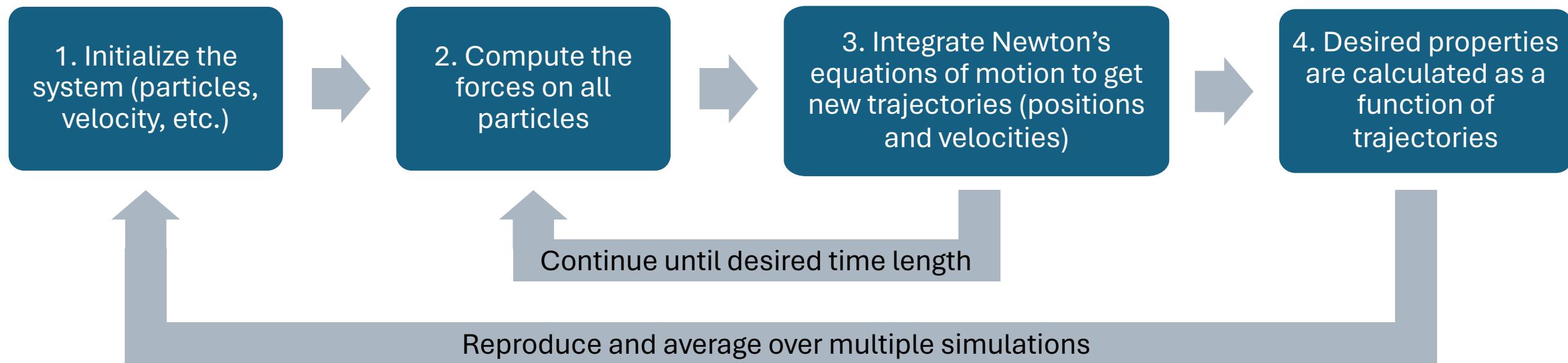
In an experiment you...



# What are the steps of MD?

Running an MD simulation is like conducting an experiment.

In MD you...



# 1. Initialize the system

To initialize a system for MD, we need to determine **the positions of each atom**, our **simulation cell**, and the **velocities of each atom**.



# 1. Initialize the system - Positions

Depending on your system of interest, there are many ways to initialize the positions of your atoms.

Online databases:

- Proteins – the [Protein Data Bank](#)
- Crystalline materials – [Materials Project](#), [Cambridge Crystallographic Data Center](#)

Initialization Software:

- [PACKMOL](#)
- [GROMACS](#)
- [VESTA](#)



# 1. Initialize the system - Positions

Once we have atomic positions, we need to minimize the structure into a ground-state configuration.

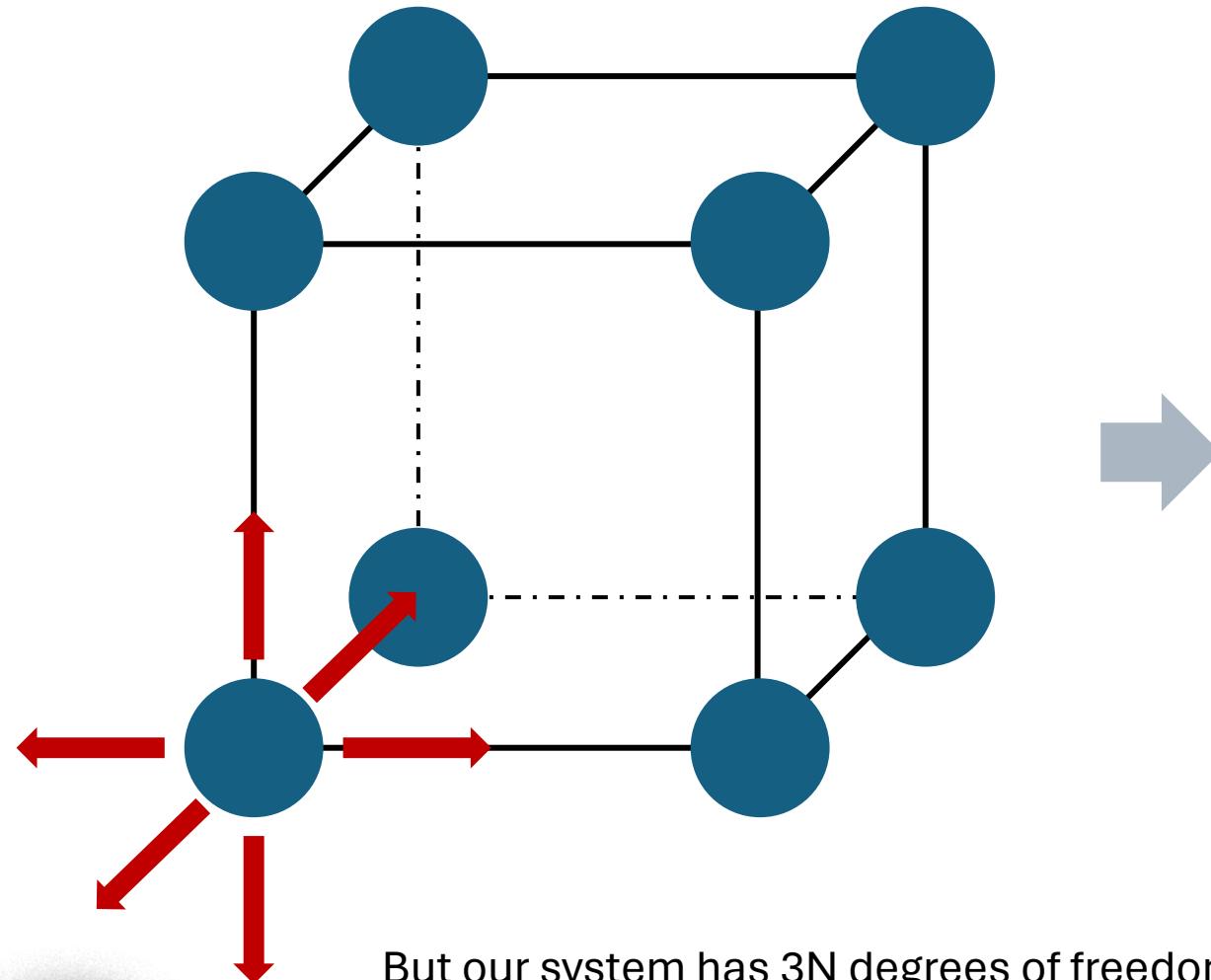
$$\frac{\partial U(x)}{\partial x} = -F(x) = 0$$

In the ground-state configuration, the derivative of the total energy of our system,  $U(x)$ , with respect to the Cartesian coordinates of all atoms will be equal to 0.

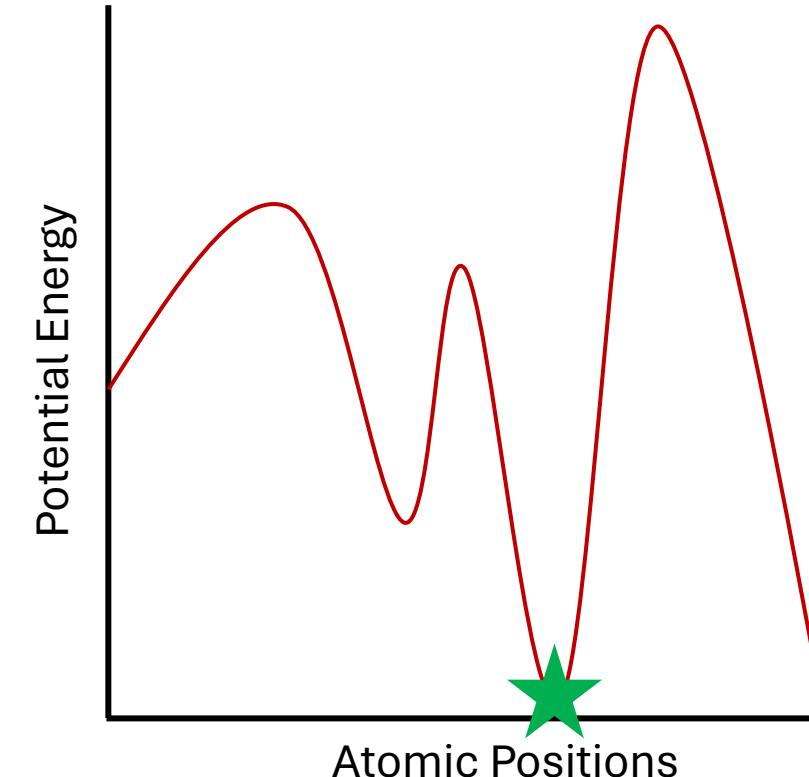
To find this energetic minimum, we must scan or **minimize** our potential energy surface by moving the positions of atoms.



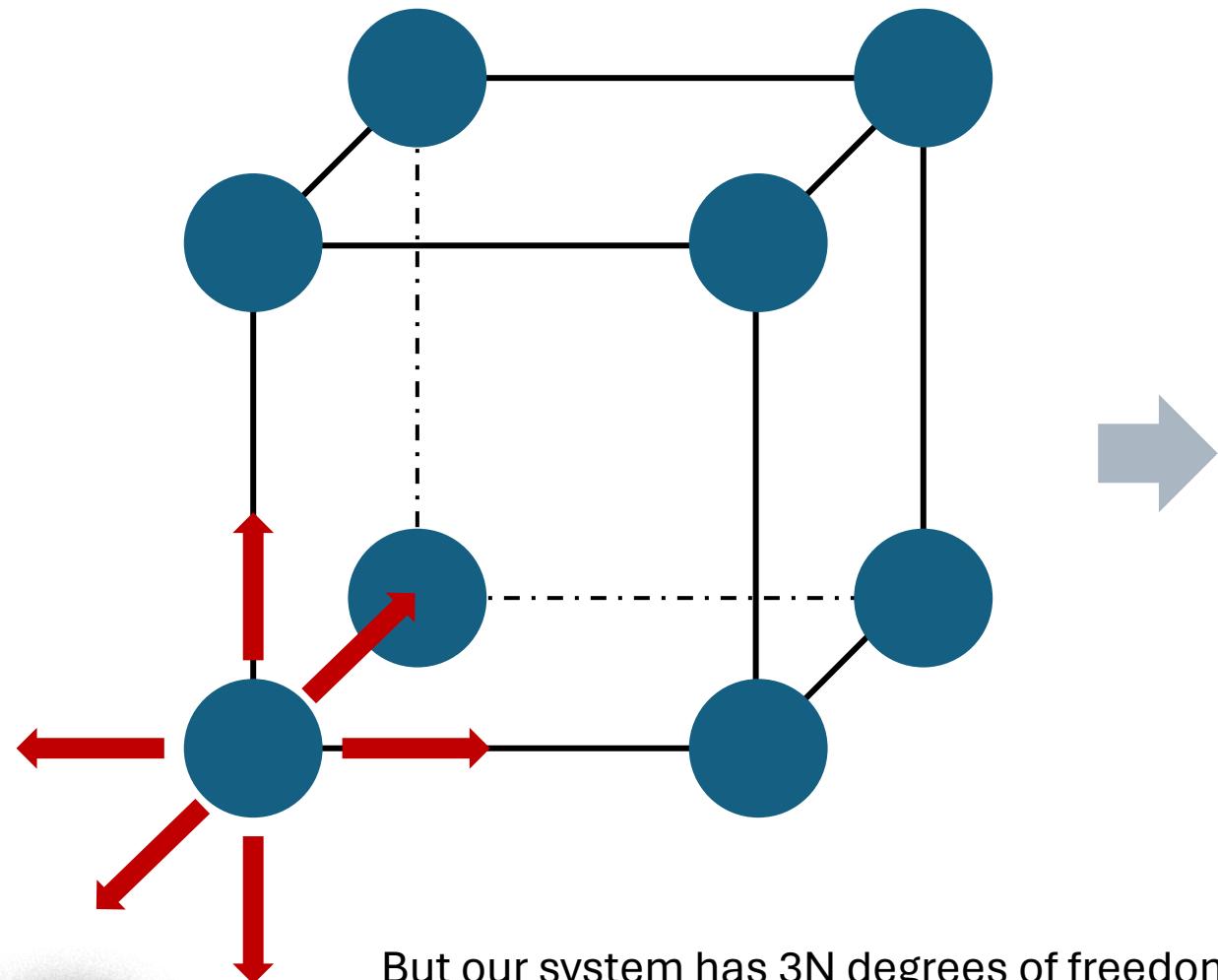
# 1. Initialize the system - Positions



But our system has  $3N$  degrees of freedom, so the potential energy surface is multidimensional...



# 1. Initialize the system - Positions



But our system has  $3N$  degrees of freedom, so the potential energy surface is multidimensional...



# 1. Initialize the system – Simulation Cell

Now that we have our atoms and their positions, we need something to contain our system in.

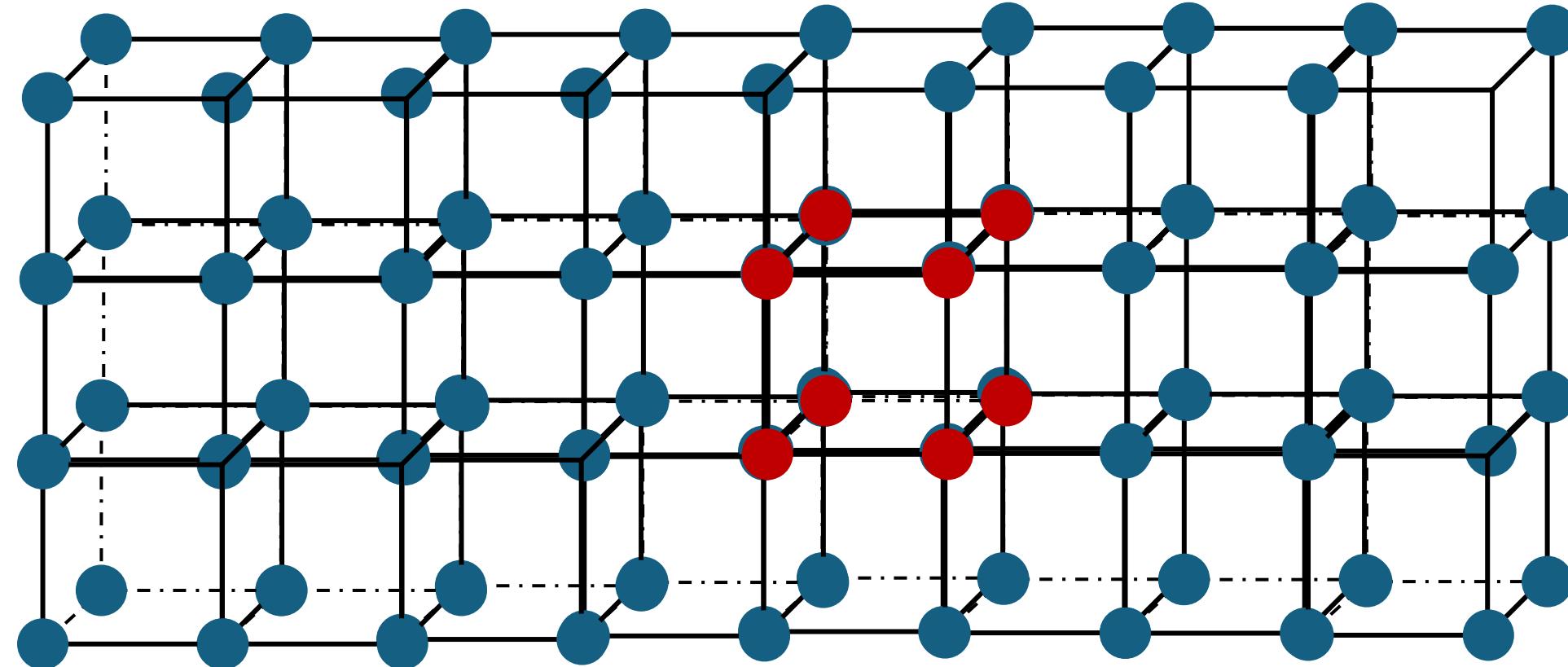
In an experiment, you might have moles of a chemical in a glass beaker, where each of those moles is on the order of  $10^{26}$  atoms.

In molecular modelling, we cannot simulate  $10^{26}$  atoms explicitly. Even if we simulate fewer atoms, the ratio of surface area to total number of atoms will always exceed reality.

We navigate this by using “periodic boundary conditions.”



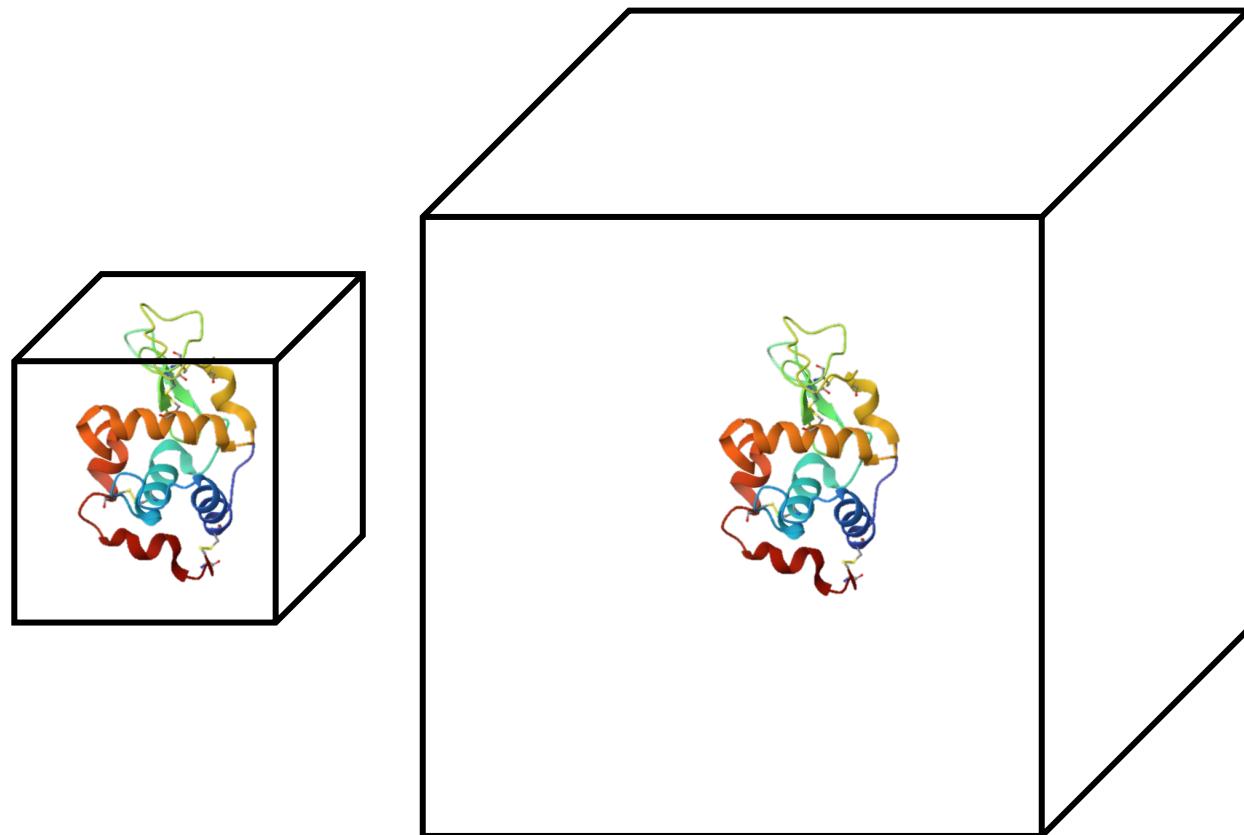
# 1. Initialize the system – Simulation Cell



But what if your system isn't crystalline?



# 1. Initialize the system – Simulation Cell



1. Don't use periodic boundary conditions. Not all modelling software support aperiodic conditions, though.
2. Be sure that your simulation box is large enough that images don't interact across the boundaries.



# 1. Initialize the system – Velocities

Initial particle velocities can be set using the Maxwell-Boltzmann distribution.

$$v_{a,i}^2 = \frac{k_B T}{m_i}$$

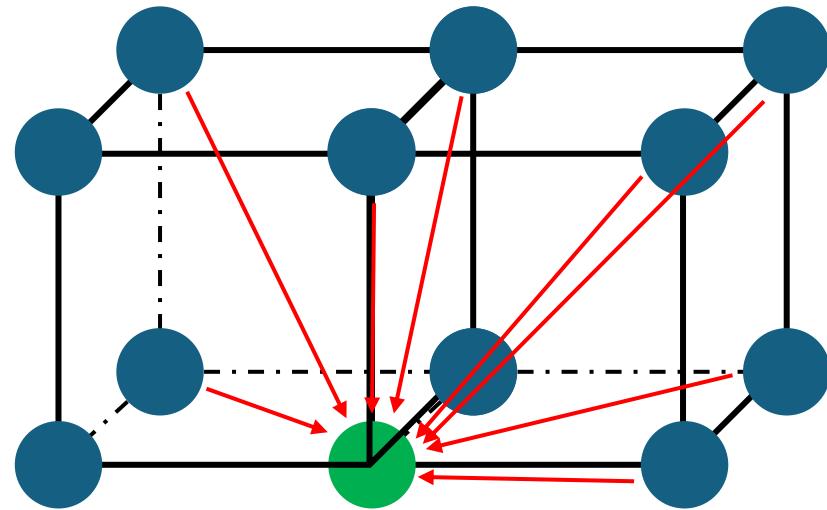
The Maxwell-Boltzmann distribution is a Gaussian distribution that utilizes this relationship to randomly set the distribution of your particle velocities at a given temperature.

A lot of software will do this for you by setting a “seed” number, which reproducibly randomizes the selection of your velocities.



## 2. Compute the forces on all particles

Calculating the forces is the most time-consuming step of a MD simulation. For each individual atom, we must consider the forces applied by all atoms around it, including the atomic images over periodic boundaries.



This quickly becomes a complicated many-body problem, particularly if we consider all the electrons on each atom. We can simplify this, however, and largely “disregard” the effect of electrons.



## 2. Compute the forces on all particles

The Born-Oppenheimer approximation is based on the principle that nuclei are much heavier than electrons.

$$F = ma$$

Since acceleration is inversely proportional to mass, nuclei are effectively static relative to electrons. The Born-Oppenheimer approximation allows us to simplify our calculation to be largely dependent on the positions of the nuclei.

However, we can't fully ignore electrons as they still have an influence over relative atomic positions.



## 2. Compute the forces on all particles

$$F_i(R^N) = -\frac{\partial U(R^N)}{\partial r_i}$$

We can calculate the forces acting on an atom by taking the first-order derivative of the potential energy, which we can get from a force-field, or potential.

$$\begin{aligned} U(R^N) &= \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] \\ &\quad + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}} \end{aligned}$$

Bonded (intramolecular) and Non-bonded (intermolecular)



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

The **bonds** between atoms are usually described by a simple harmonic equation. However, this inherently implies that a bond cannot be broken, so most force fields are *not* reactive. Reactive force fields exist but be sure to read their literature before use.



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

The **angles** between bonds are also usually described by a simple harmonic equation.



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

**Torsions**, or dihedrals, are the bond angles between four or more atoms in a row. These properties are important for replicating things like conformational changes. “Improper” dihedrals account for change in planarity, such as sp<sup>3</sup> hybridization or aromatic rings.



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

The **van der Waals** terms account for the attractive and repulsive forces between atoms. The form currently shown is the 12-6 Lennard-Jones potential. This is used commonly for speed, but it is not the most accurate form available.



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

The final term accounts for the **Coulomb** interactions, or electrostatics, between electrons. It is very common for this term to be calculated from partial atomic charges from *ab initio* methods. Force fields differ quite a bit on this term, though, as far as how far electrostatics are truncated, where the atomic charge is placed, or how they are derived.



## 2. Compute the forces on all particles

$$U(R^N) = \sum_{Bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{Angles} \frac{1}{2} k_a (\theta - \theta_0)^2 + \sum_{Torsions} \frac{V_n}{2} [1 + \cos(n\phi - \delta)] + \sum_{Improper} V_{imp} + \sum_{LJ} 4\epsilon_{ij} \left( \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^6 \right) + \sum_{Coulomb} \frac{q_i q_j}{r_{ij}}$$

Additional terms can be added to the force field form to increase accuracy. Commonly added terms include corrections for hydrogen bonding or polarization. Be sure to thoroughly read literature to find accurate force fields to use if these effects happen in your system.



## 2. Compute the forces on all particles

Deriving force fields is complicated and usually the focus of entire dissertations. Unless this *is* your dissertation work or you have a *very* specific system, you will typically use ones readily available.

1. What effects are important in your system? Hydrogen bonding? Polarizability? Accurate van der Waals? Accurate electrostatics? Does dynamic bonding need to be captured?
2. Read the literature!
  1. Are there other papers studying similar systems? Similar effects? What force fields did they use?
  2. Read the publications for the force fields to determine what systems they're applicable to and how accurate they are, what their limitations might be.



## 2. Compute the forces on all particles

Universal force fields:

- Dreiding
- Universal force field (UFF)

Reactive force fields:

- ReaxFF

Popular biological force fields:

- CHARMM
- AMBER
- GROMOS

Water

- TIP series (TIP3P, TIP4P...)
- SPC-E

Machine learning developed force fields



## 2. Compute the forces on all particles

Important force field limitations:

- Cannot handle dynamic bonding, charge transfer, electron excitation, polarizability, etc.
- Accuracy heavily depends on the quality of the data that was used to develop parameterization. Results should be validated with experimental work where possible.
- Transferability is limited. Force fields are usually parameterized for specific atoms, temperature and pressure ranges, etc.



### 3. Integrate Newton's equations of motion

$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i = -\frac{\partial U(R_i)}{\partial r_i}$$

The left-side equation is a second-order non-linear differential equation. This must be solved numerically using an integration algorithm. To do this, we will discretize our solutions.

$$r_i(t_0) \rightarrow r_i(t_0 + \Delta t) \rightarrow r_i(t_0 + 2\Delta t) \rightarrow \dots r_i(t_0 + n\Delta t)$$

$\Delta t$  is the timestep of our simulation. This determines how frequently we “sample” our system to calculate the forces and new trajectories. If we do this for  $n$  number of steps this will yield our simulation length.

$$n \cdot \Delta t = \text{length}$$



# 3. Integrate Newton's equations of motion

A good integrator must...

- Have minimal dependence on the forces.
- Have good stability for larger timesteps.
- Have good accuracy.
- Conserve momentum and energy.
- Be time-reversible.

There are three common integrators: Verlet, leap-frog, and velocity Verlet.



### 3. Integrate ... – Verlet Integrator

$$r_i(t_0 + \Delta t) = -r_i(t_0 - \Delta t) + 2r_i(t_0) + \frac{d^2 r(t_0)}{dt^2} \Delta t^2 + \mathcal{O}(\Delta t^4)$$

The error of this integrator is on the order of  $\mathcal{O}(\Delta t^2)$ . Notice, also, that this is only dependent on the positions and not velocities.

$$v_i(t_0) = \frac{1}{2\Delta t} [r_i(t_0 + \Delta t) - r_i(t_0 - \Delta t)]$$

The error of our velocities, which we use to calculate instantaneous temperature, is on the order of  $1/\Delta t$ , meaning that at small timesteps, this error is very large.



# 3. Integrate ... – Leap-frog Integrator

$$r_i(t_0 + \Delta t) = r_i(t_0) + v_i\left(t_0 + \frac{\Delta t}{2}\right)\Delta t$$
$$v_i\left(t_0 + \frac{\Delta t}{2}\right) = v_i\left(t_0 - \frac{\Delta t}{2}\right) + \frac{d^2 r(t_0)}{dt^2} \Delta t$$

The instantaneous atomic velocity is calculated as...

$$v_i(t) = \frac{1}{2} v_i\left(t + \frac{\Delta t}{2}\right) + \frac{1}{2} v_i\left(t - \frac{\Delta t}{2}\right)$$

Our error is now proportional to our timestep, so for sufficiently small timesteps, we can reduce our error significantly.

Now that our equations of motion depend on velocity, we can couple our system to an external bath, allowing energy to vary.



### 3. Integrate ... – Velocity-Verlet Integrator

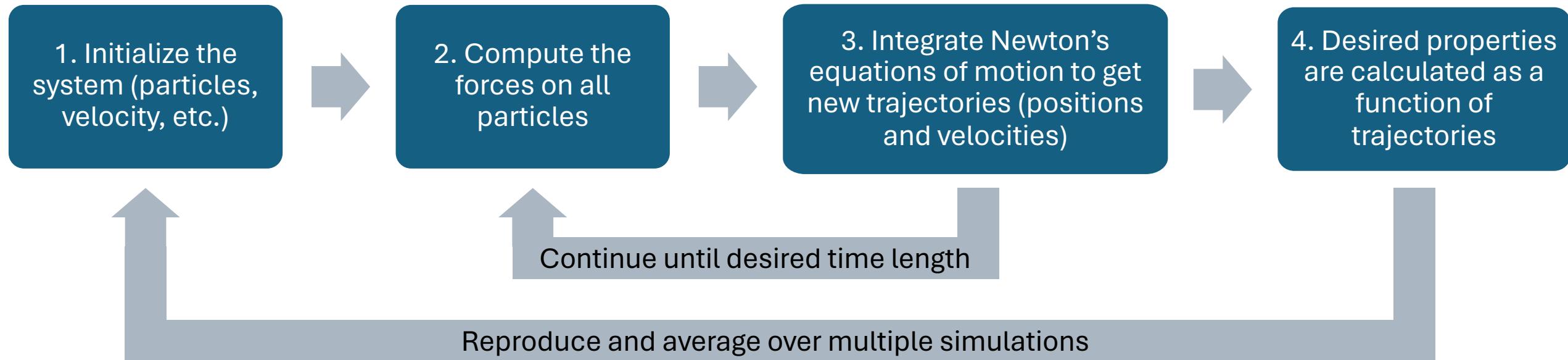
$$r_i(t_0 + \Delta t) = r_i(t_0) + v_i(t_0)\Delta t + \frac{1}{2} \frac{d^2r(t_0)}{dt^2} \Delta t^2$$
$$v_i(t_0 + \Delta t) = v_i\left(t_0 + \frac{\Delta t}{2}\right) + \frac{1}{2} \frac{d^2r(t_0 + \frac{\Delta t}{2})}{dt^2} \Delta t$$

The velocity-Verlet algorithm is superior to the leap-frog algorithm. The computational speed has been improved by simultaneously calculating the velocity at  $(t_0 + \Delta t)$ .

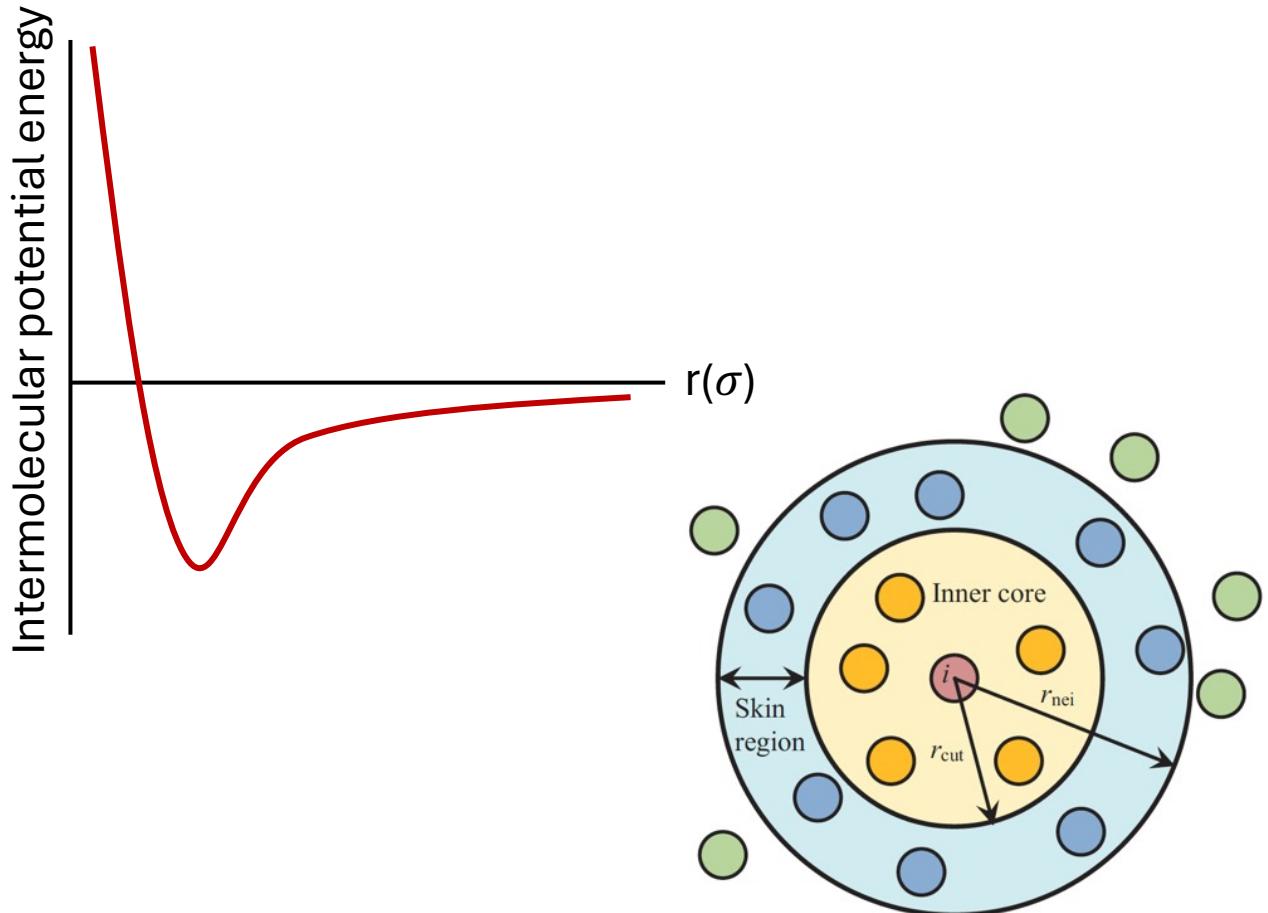
$$v\left(t_0 + \frac{\Delta t}{2}\right) = v(t) + \frac{1}{2} \frac{d^2r(t_0)}{dt^2} \Delta t$$



# 4. Repeat



# 4. Repeat



Zhou and Liu, 2022.

At the end of each iteration, we must update our neighbor list for each atom.

Generally, we can truncate the list to end after a certain “cut-off” distance since non-bonding interactions decrease with distance.

A common neighbor list algorithm was also developed by Verlet, which gives a cut-off distance and “buffer” zone.



# Ensembles

Typically, in MD, we are not interested in the properties of each atom, but rather the property of our entire system or a collection of atoms.

We assume in MD that our systems are **ergodic** meaning that a single particle will experience the same states over time as a group of particles at a discrete time. The average property of our particles is then the same as the ensemble average of our system.



# Ensembles

There are three commonly used ensembles:

- Canonical (NVT)
- Microcanonical (NVE)
- Isothermal-isobaric (NPT)

Each ensemble is defined by the thermodynamic properties that are held constant over the specified length of time.



# Ensembles - Canonical

Properties held constant:

- Number of particles, N
- Volume, V
- Temperature, T

Our system is not isolated from its surroundings. Energy must be exchanged to maintain the temperature. This is done by using a **thermostat**.

There are a variety of thermostat algorithms available, and they range in accuracy. Again, read the literature!



# Ensembles - Microcanonical

Properties held constant:

- Number of particles, N
- Volume, V
- Total energy, E

Our system *is* isolated from its surroundings to keep the energy constant.

$$E = E_{kinetic} + U$$

While the total energy, E, is kept constant, the kinetic and potential energy can fluctuate. This can, however, lead to large temperature increases, so use carefully.



# Ensembles – Isothermal-isobaric

Properties held constant:

- Number of particles, N
- Pressure, P
- Temperature, T

As with the canonical ensemble, our system is not isolated from its surrounding. To maintain the pressure, we can use a **barostat**.

Again, read the literature on the accuracy of different barostats before using in your research!



# Ensembles

How do you decide what ensemble you should use?

- What are you simulating?

A gas-phase system can be done using an NVE or NVT ensemble. A liquid-phase system will be done using NPT to maintain the correct density.

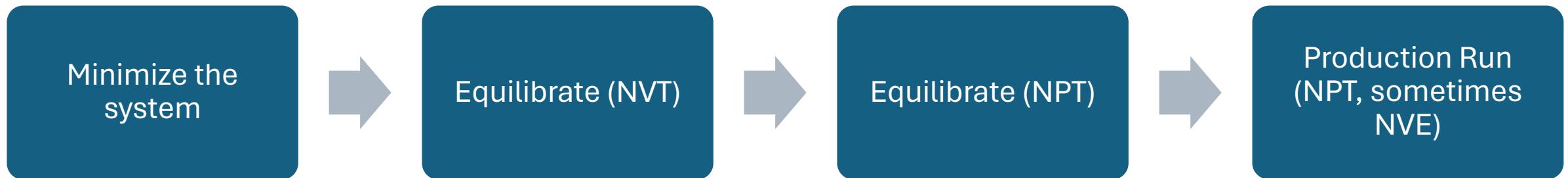
- What is the comparable experiment?

Many experiments are done at a constant pressure, so the NPT ensemble is commonly used.

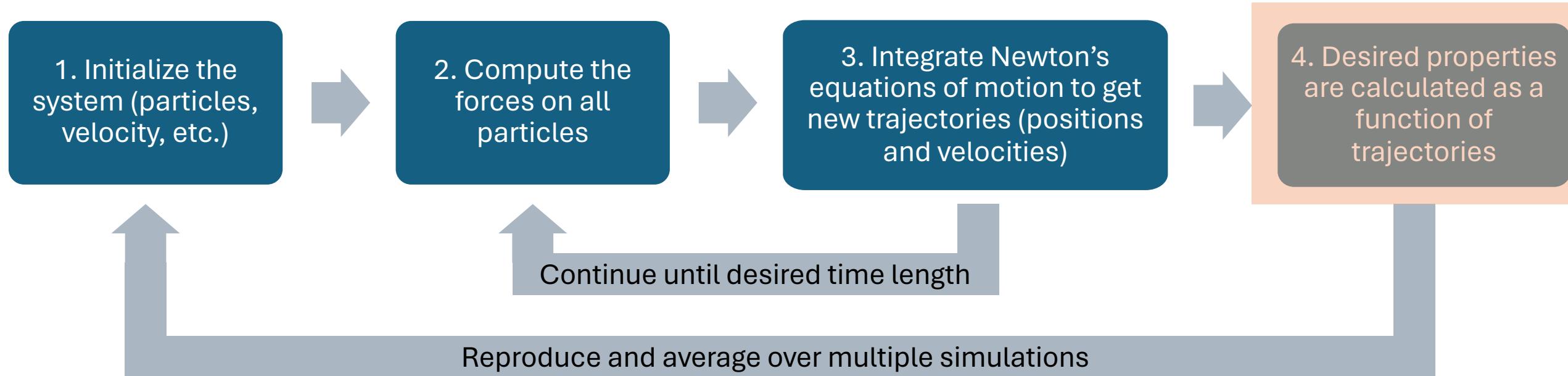
- Are you interested in calculating free energies?
  - NVE – internal energy
  - NVT – Helmholtz free energy
  - NPT – Gibbs free energy



# A Typical MD “flow”



# 4. Repeat



# How do we calculate our desired properties?

Typically, our properties are calculated as a function of position, velocity, or force per atom.

Temperature

$$T(t) = \frac{2K(t)}{3N-3}$$

Radial Distribution Function

$$g(r) = \frac{dN}{4\rho_N \pi r^2 dr}$$

Diffusion coefficient

$$\lim_{t \rightarrow \infty} \langle |r_i(t) - r_i(0)|^2 \rangle = 6Dt$$

Correlation functions

$$C(m\Delta t) = \frac{1}{n} \sum_{i=0}^{n-1} A(m\Delta t + i\Delta t)A(i\Delta t)$$



# Important limitations of molecular dynamics

- Not reactive since we do not explicitly account for dynamic behavior of electrons.
- The accuracy of your calculations is heavily dependent on the accuracy and applicability of the force field.
- Errors can also result from your integrator, thermostat, barostat, and timestep.
- Time and size limitations.



# Example Problem



# Introduction to Density Functional Theory



# What is density functional theory (DFT)?

DFT is a **quantum mechanical** computational modelling method used to investigate the electronic-structure of many-body systems.

DFT is a type of *ab initio* calculation as it attempts to solve the Schrödinger equation to determine the electronic structure.

In contrast, MD looks at transport properties, meaning time-dependent properties and how they change over time. DFT is generally a single-point method that calculates the properties of a ground-state structure.



# Schrödinger's Equation (not the cat)

$$\hat{H}\Psi(r_1, r_2, \dots r_N) = E\Psi(r_1, r_2, \dots r_N)$$

“Where did we get that from? It’s not possible to derive it from anything, you know. It came out of the mind of Schrödinger.”

- Richard Feynman



# Schrödinger's Equation (not the cat)

Time-independent Schrödinger equation.

$$\hat{H}\Psi(r_1, r_2, \dots r_N) = E\Psi(r_1, r_2, \dots r_N)$$

From a quantum perspective, electrons have particle-wave duality. Therefore, we need a function to describe this behavior in space, or in a given orbital. This is known as the wavefunction,  $\Psi$ .

The absolute value of the squared wavefunction tells us the probability of finding an electron around an atom. This is known as the electron density.

$$\rho(r) = \sum_{i=1}^N |\psi(r_i)|^2$$

# Schrödinger's Equation (not the cat)

$$\hat{H}\Psi(r_1, r_2, \dots r_N) = E\Psi(r_1, r_2, \dots r_N)$$

The left-hand term,  $\hat{H}$ , is known as the Hamiltonian operator. This is a set of functions that when applied to the wavefunction will give us the right-hand side of the equation, which is the total energy of our system, E.

The Hamiltonian is the sum of kinetic energy, electrostatic energy from an external potential, and electron-electron interaction energy.

$$\hat{H} = -\frac{1}{2} \sum_i^N \nabla_i^2 + \hat{V}_{ext} + \sum_{i < j}^N \frac{1}{|r_i - r_j|}$$

This cannot be numerically solved for anything other than Hydrogen.

# Schrödinger's Equation (not the cat)

$$\hat{H}\Psi(r_1, r_2, \dots r_N) = E\Psi(r_1, r_2, \dots r_N)$$

To simplify the solution, we will again use the Born-Oppenheimer approximation to separate electron and nuclear wavefunctions.

$$\hat{V}_{ext} = - \sum_{\alpha=1}^M \frac{Z_\alpha}{|r_i - R_\alpha|}$$

Using the Born-Oppenheimer approximation,  $\widehat{V}_{ext}$  depends only on the charge of the nucleus ( $Z_\alpha$ ) and its separation from the electron.

Even though we have simplified the wavefunction, the problem becomes impossible as we add more electrons, so further approximation is necessary.



# Schrödinger's Equation (not the cat)

$$\rho(r) = \sum_{i=1}^N |\psi(r_i)|^2$$

Hohenberg-Kohn Theorem 1:

*The ground-state energy of a system can be calculated as a functional of the electron density.*

Hohenberg and Kohn were able to prove through derivation that this is true, but they could not prove the exact form of the functional that is required.



# Schrödinger's Equation (not the cat)

$$\rho(r) = \sum_{i=1}^N |\psi(r_i)|^2$$

Hohenberg-Kohn Theorem 2:

*The electron density that minimizes the energy of the functional is the true electron density corresponding to a full solution of the Schrödinger equation.*

Again, though, this functional form is not known.

Also, this theorem implicates that DFT is effectively limited to ground-state systems.



# Kohn-Sham Equations

So, how do we actually *solve* the Schrödinger equation?

Kohn-Sham proposed a set of equations where each equation is dependent only on a single electron wavefunction.

$$\left[ -\frac{\hbar^2}{2} \nabla^2 + V(r) + V_H(r) + V_{XC}(r) \right] \psi_i(r) = E \psi_i(r)$$

The first two terms are known; this is the kinetic energy and the external interaction energy between an electron and the nuclei.

The third term is the Hartree potential, which describes the interaction of the single electron with the whole electron density.



# Kohn-Sham Equations

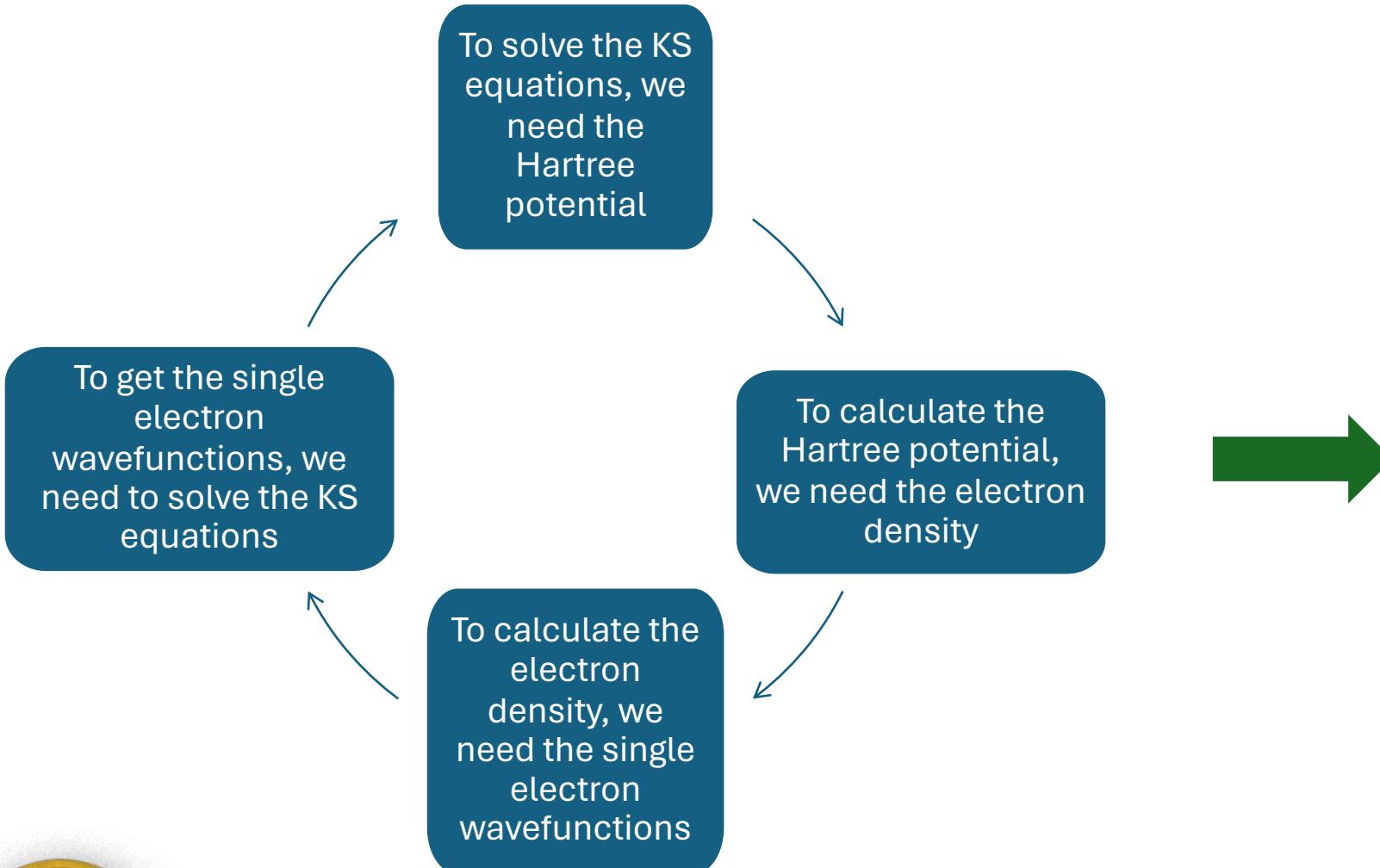
So, how do we actually *solve* the Schrödinger equation?

Kohn-Sham proposed a set of equations where each equation is dependent only on a single electron wavefunction.

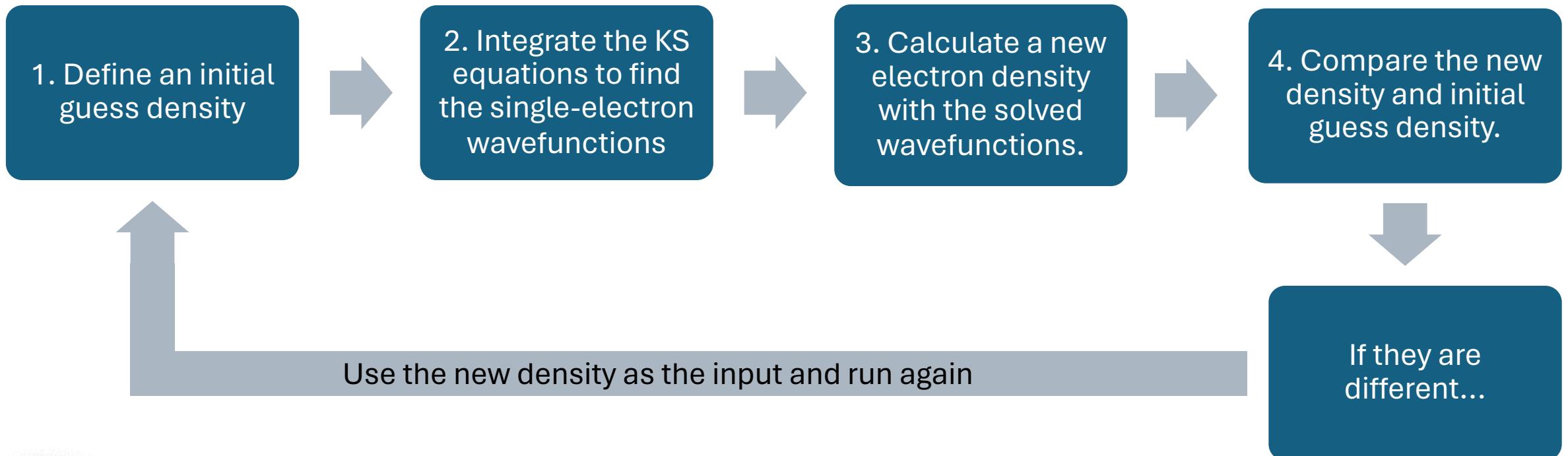
$$\left[ -\frac{\hbar^2}{2} \nabla^2 + V(r) + V_H(r) + V_{XC}(r) \right] \psi_i(r) = E \psi_i(r)$$

The final term is the exchange-correlation energy. This accounts for the errors made in assuming that we have non-interacting single-electron wavefunctions.

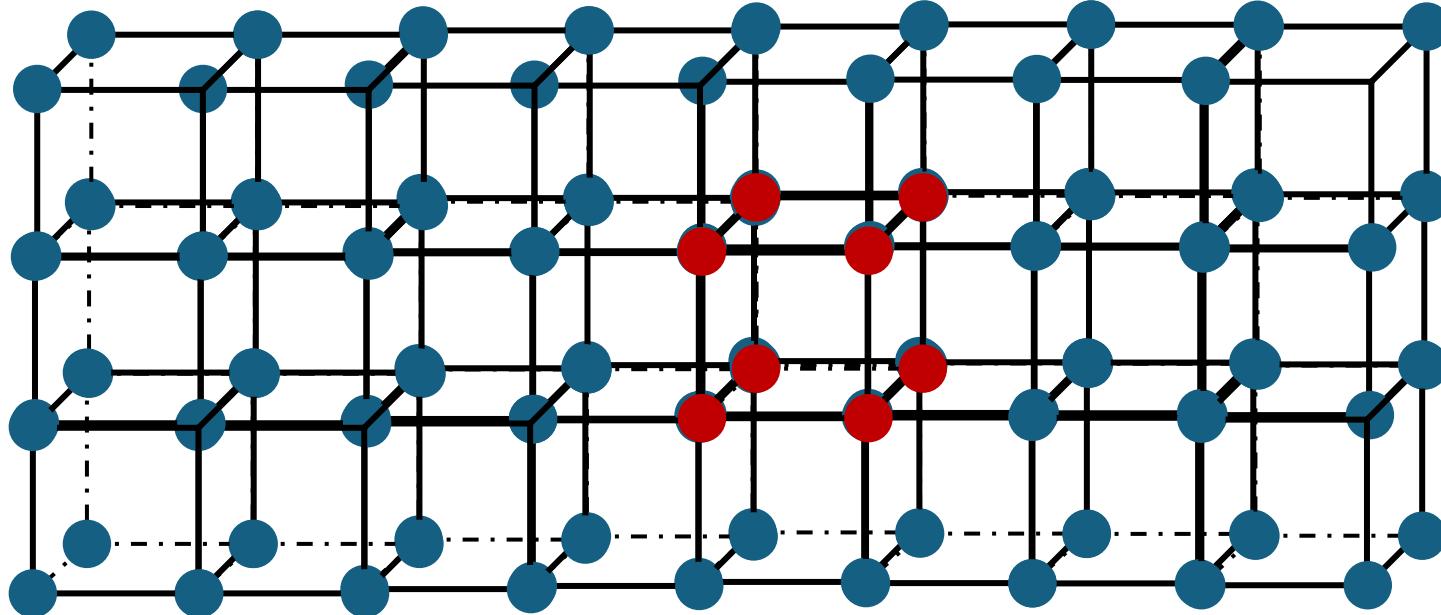
# Kohn-Sham Equations



# Self-consistent DFT



# Returning to periodic boundary conditions...



Bloch's theorem is maintained in classical MD as well. However, it's only of explicit concern to you as the user if you are calculating wave properties, such as phonon transport.



# Minimization, again

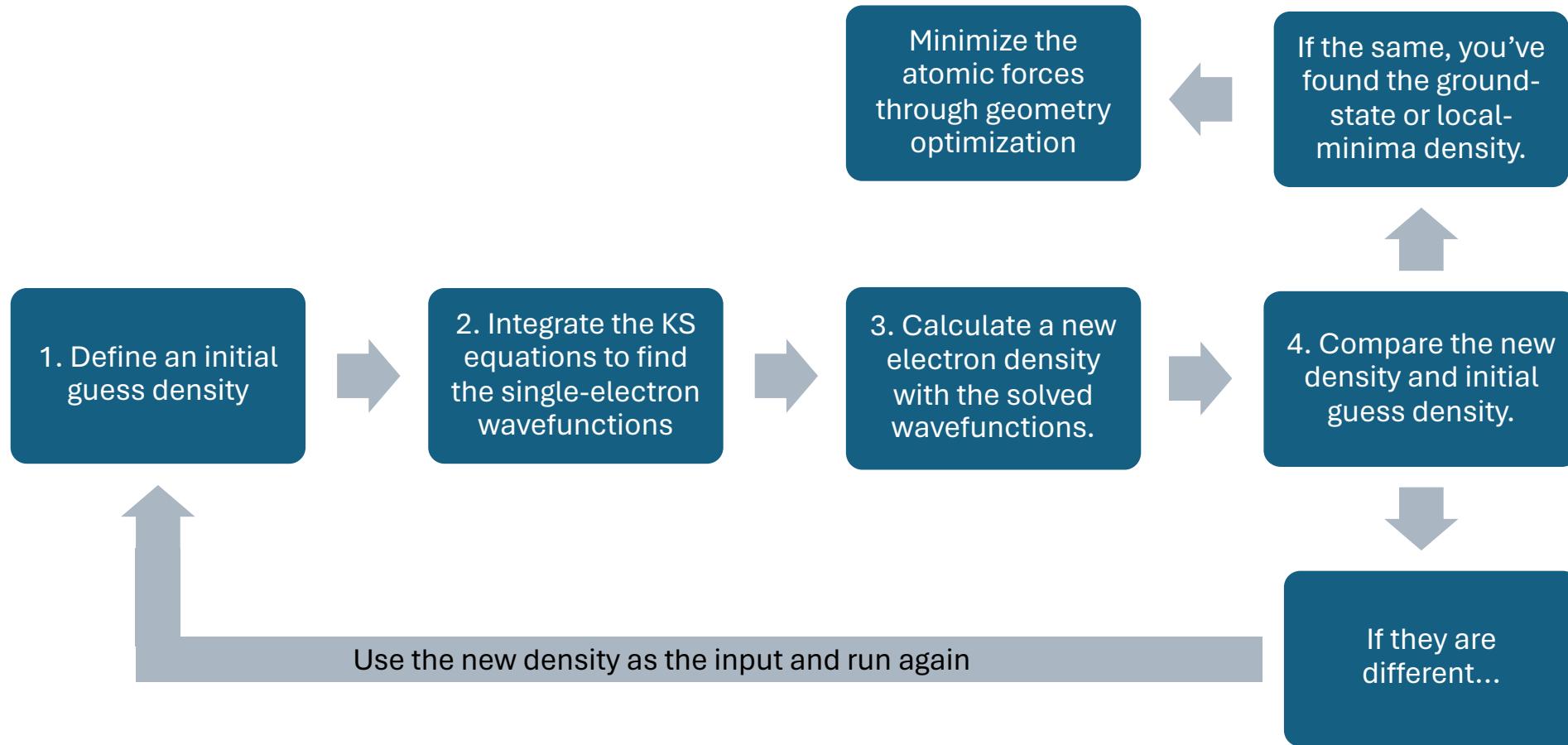
While we iteratively determine the electronic structure, we also care about the location of our atoms.

1. Shift the position of atoms with respect to one another and minimize the forces
2. Change the lengths of our simulation cell, which affects periodicity

A starting DFT calculation can also be called a **minimization** or **geometry optimization**.



# Minimization, again



# What is the user input to a DFT calculation?

1. Initializing your system
2. Optimization algorithm
3. What functional to use
4. What pseudopotential to use
5. K-points (how thoroughly we sample our electronic structure)
6. The convergence criteria (for forces and electron density)



# What are the important limitations of DFT?

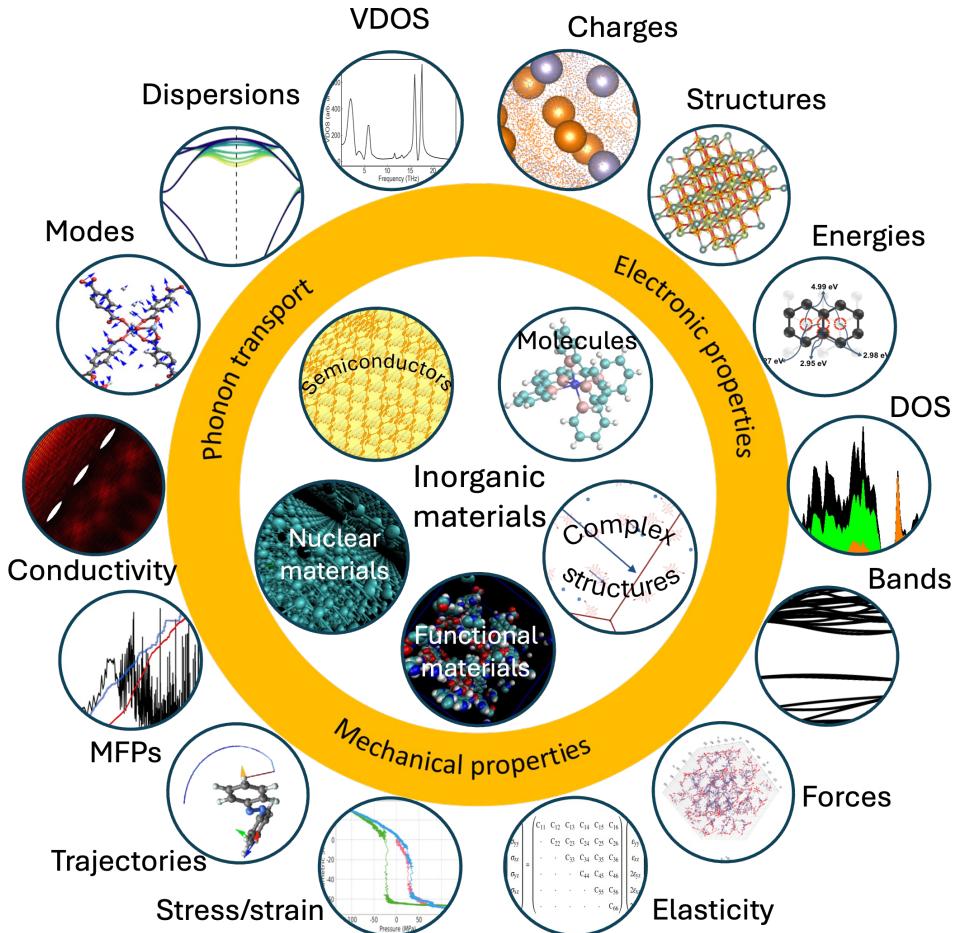
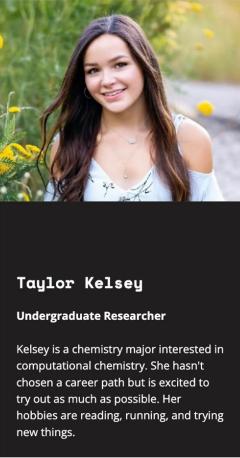
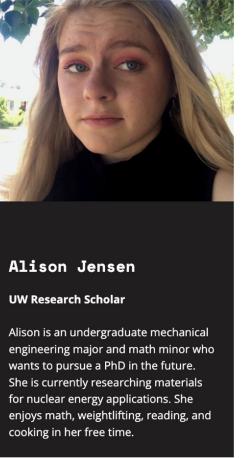
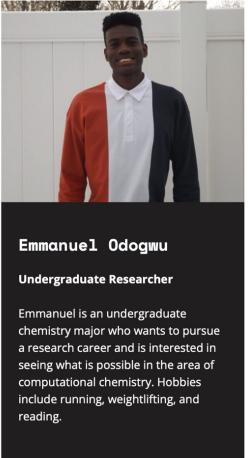
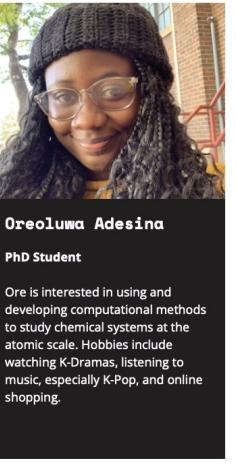
- DFT does not truly solve the Schrödinger equation
- It is difficult—but not impossible—to model excited electron states
- DFT underpredicts the band gap of semiconductors
- Inaccurate descriptions of van der Waals interactions
- *Very limited system size, and time when doing *ab initio* MD*



# DFT Demonstration – Partial Atomic Charges



# Computational Design of Inorganic Materials

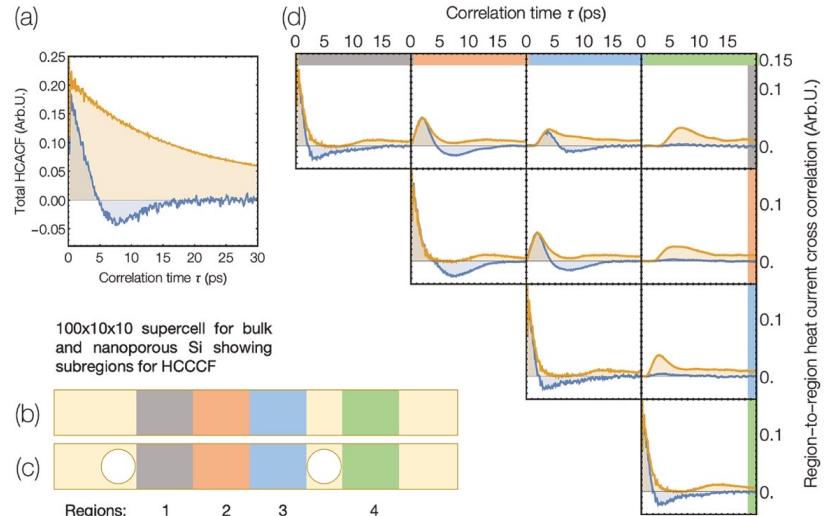


06/12/2024

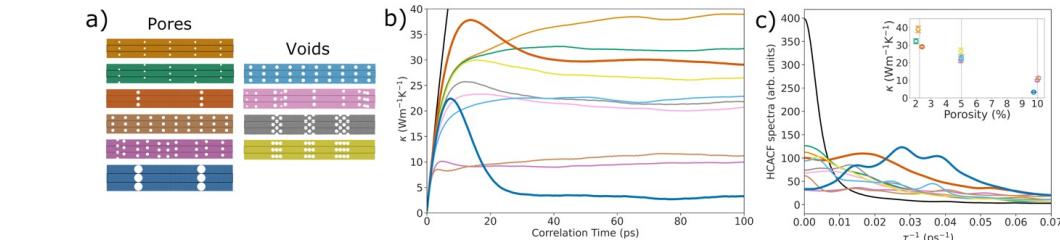
Bio-Comp Hackathon 2024

65

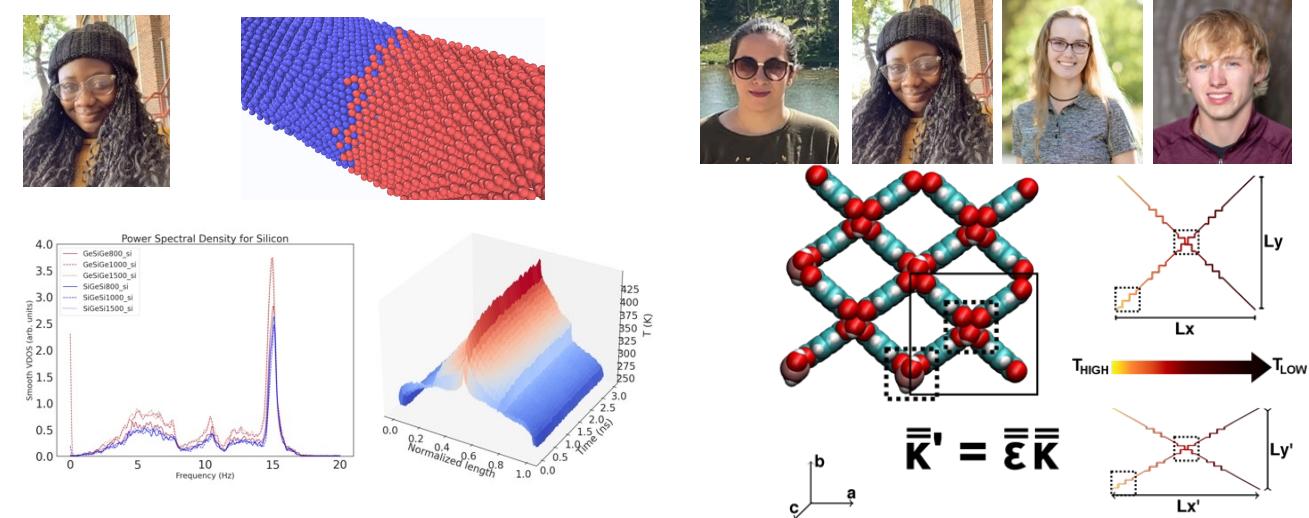
# Thermal Transport via MD



Hosseini, Davies, Dickey, Neophytou, Greaney, **de Sousa Oliveira**, Materials Today Physics, Volume 27, 2022, 100719

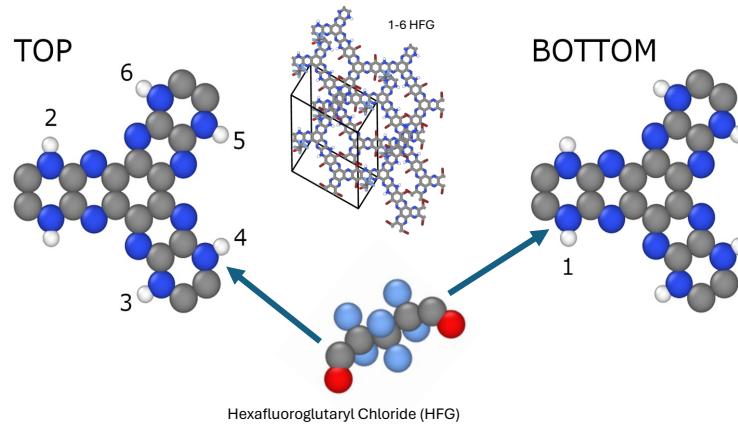


Greaney, Hosseini, Davies, de Sousa Oliveira, Neophytou, Nanomaterials, Volume 14, 2024, 795

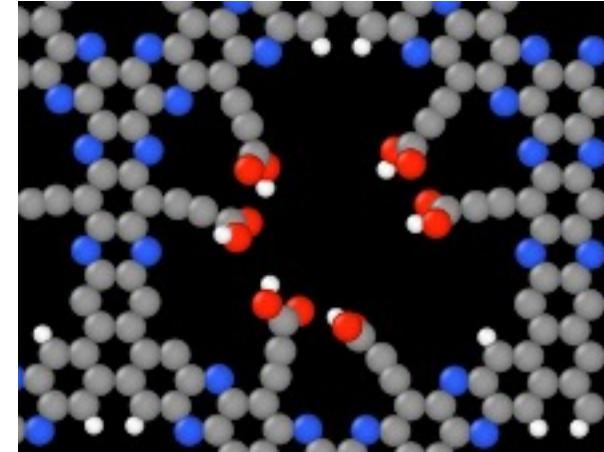


# Stability and Semiconducting via DFT

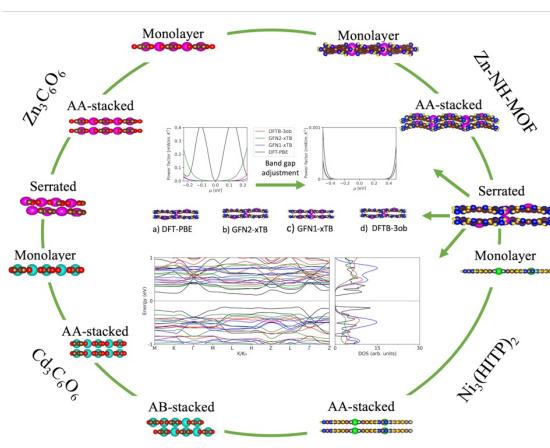
Increased stability via cross-linking (COFs)



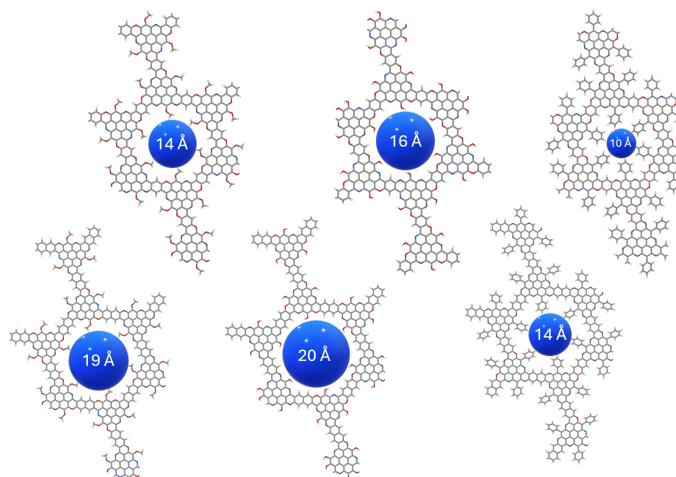
Functional group degradation (COFs)



Semiconducting MOFs



Semiconducting COFs



# Useful Resources



## Online Resources:

- [MD Tutorials](#)
- [GROMACS](#)
- [Protein Data Bank](#)

## Textbooks:

- Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications: Third Edition*. 2023.
- Kun Zhou and Bo Liu. *Molecular Dynamics Simulation: Fundamentals and Applications*. 2022.
- David Sholl and Janice Steckel. *Density functional theory: A practical introduction*. 2009.
- Sizhe Liu and Binquan Luan. *Benchmarking various types of partial atomic charges for classical all-atom simulations of metal–organic frameworks*. *Nanoscale*, 14(26), 2022.

