Application of Demon Algorithm in the Study of Phonon Transport

Rithik
Saratoga High School
(Dated: September 27, 2025)

The demon algorithm has been applied in the study of phonon transport with the intent of understanding heat conduction in nanoscale devices. Initially starting with an ideal gas system, we adapted the demon algorithm to model Einstein Solids effectively. Such Monte Carlo simulations serve as efficient methods to simulate the behavior of heat dissipation on quantum scales, where energy remains as discrete quantities. The exploration and adaptation of the modified demon algorithm is explored in this study.

I. INTRODUCTION

In a world where everything continues to grow smaller, it becomes increasingly difficult to compute and understand the transfer of energy. Small-scale interactions differ greatly with that of larger classical models, and involve considering more parameters and values. Specifically, we will explore energy transfer in quantum systems where energy takes on discrete properties, such as on a nanoscale. Systems are unable to be modeled through classical continuous energy models, and thus must be looked at differently. Utilizing Monte Carlo simulations allows for advanced simulations on the physics of particles and thermodynamic system modeling. We can apply these critical simulations to help model the situation, adhering to specific conditions such as fixed energy and more.

With more recent advances in technology, there has been a trajectory towards significantly smaller technology, such as nanodevices. With technology like microprocessors or quantum dots, this small scale revolution has started to move towards more commercial use, including the mobile phone today. However, studies of heat transfer become increasingly critical to understand, with build up of excessive heat causing extensive damage or unwanted behavioral modifications to nanodevices. To prevent energy loss and sustain optimal device performance involves heat transfer to occur efficiently. This scale involves discrete energy, with energy being carried by carriers on the quantum scale, such as phonons. These heat carries travel through ballistic transport, by moving without scattering in areas to optimize transport. Through efficient heat dissipation, the buildup of heat is avoided which in turn allows for the system to continue operating efficiently.

We are able to study particulate behavior on these scales to better work and optimize nanoscale physical simulations through modeling. Molecular dynamics allows for the physical movement of molecules and atoms to be studied, in accordance to Newton's equations of motions. Similarly, Monte Carlo simulations, for example, serve as another way to model thermodynamic systems and particle behavior through random sampling and other methods. In this study, we utilize a Monte Carlo method known as a Demon Algorithm to sample particles in a microcanoncial ensemble to better study vibrations of atomic lattices, where heat is generated and measured. By avoiding brute force methods, we can efficiently work with such quantum systems with this algorithm, to draw conclusions on thermal behavior, entropy, and equilibrium of phonon based systems.

II. THE MICROCANONICAL ENSEMBLE

An ensemble represents a large number of imaginary replicas of a physical system. Certain ensembles can contain large numbers of microelements referred to as particles. Systems with particles can demonstrate different configurations of particles, known as microstates, while adhering to specific criteria. Though particles exist on microscopic scales, the criteria that apply to these systems are macroscopic, on a larger scale, and often involve having certain properties to be constant. These macroscopic criteria exist in various forms, such as fixed energy or volume, placing heavy constraints on the systems as a whole. Conversely, microstates depict specific combinations and organization of particles based on their positional probability. However, when the system is in equilibrium, under the same physical constraints placed upon the system, every microstate, such as different arrangements of particles, can be assumed to be equally likely as there is no reason to select one over the other.

A microcanonical ensemble is an ensemble of systems in which the total energy is specified, and the number of particles and the volume is a constant. One of its applications is understanding the equilibrium state of a system, and relies on the the postulate of a priori equal probabilities which states that all microstates are equally likely to occur at equilibrium. It uses a quantity known as *density of states* denoted by Ω which is the total number of possible microstates within the constraints of constant energy E, number of particles N, and volume V. In classical systems, the microstates occur in the form of a continuous distribution, while in quantum systems they are discrete. Consequently, we can express Ω for classical systems as:

$$\Omega = \int_{accessable} \{dp\ dq\} = \Omega(E, V, N) \tag{1}$$

where $\{dp\ dq\}$ represents the total elemental volume in momentum and position space of all the particles, also known as phase space volume. The integration is carried out over all accessible states within the constraints of constant energy E, number of particles N, and volume V. In quantum systems this expression is represented as a summation over all accessible states, and is given by:

$$\Omega = \sum_{accessable} [\ 1\ ] = \Omega(E, V, N) \tag{2}$$

In classical systems, Ω is the accessible volume of phase space. In quantum systems, it is the total number of

microscopic states accessible to the system. The former can be made dimensionless by using a normalization factor, while the latter is naturally dimensionless.

The microcanonical ensemble is based on the postulate that the system is *equally likely* to be in any of its accessible microstates. To make this postulate of *equal a priori probabilities* may be represented as:

$$P_s = \begin{cases} 1/\Omega, & \text{if s is accessible} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where the sum of Ps over all Ω is equal to unity. This equation is applicable only when the system is in equilibrium.

A. The Microcanonical Ensemble Interpretation of Temperature and Entropy

Microcanonical ensembles can be used as a way to describe the definition of temperature. We can start by considering two systems that are isolated but can interact with each other through thermal energy transfer. We assume that a partition exists between the two systems in a way that energy can easily flow through them, while overall energy of both systems combined remains fixed, as shown in Fig. 1. As energy can flow between the two systems, the energy of each system is not fixed individually.

We can denote this relationship between energies as

$$E_0 = E + E' \tag{4}$$

where E is the energy of system 1 and E' is the energy of system 2, and E_0 is the total energy.

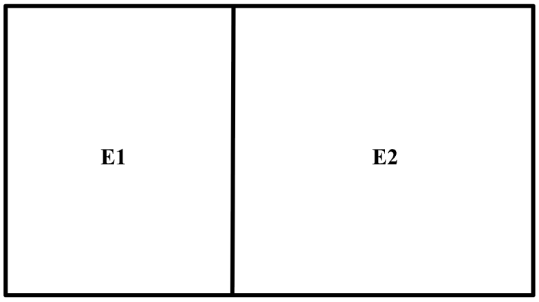

FIG. 1. An illustration of two systems with a divider in between. Energy is able to flow between the systems, which both remain in isolation from everything else.

Based on the postulate of a priori equal probabilities, every microstate has an equally likely chance of existing at equilibrium. Therefore, we can state accordingly that the probability $P(E)$ of finding a microstate of the system is directly proportional to the total number of microstates available to the system. This assumption, stemming from equal probability of finding any microstate, can be written with respect to C the constant of proportionality which is independent of energy. $\Omega^{(0)}$ represents number of states accessible to the total system, and is related to $P(E)$ as:

$$P(E) = C\Omega^{(0)}(E) \tag{5}$$

We can also suppose that the possible states accessible to System 2 can coincide with any of the possible states accessible to System 1, yielding the total number of possible states of the entire system as:

$$\Omega^0(E^0) = \Omega(E)\Omega'(E^0 - E) \tag{6}$$

where $\Omega(E)$ represents the density of states of System 1 with energy E, and $\Omega'(E^0 - E)$ that of System 2 with energy $(E^0 - E)$, and we have used the fact that sum of the energies of the two systems is E^0. Thus, we have:

$$P(E) = C\Omega(E)\Omega'(E^0 - E) \tag{7}$$

The functions Ω and Ω' represent all the possible microstates for a given energy. Since these are macro systems, they will contain a very large number of microstates. For example, the number of molecules in one mole of an ideal gas is

of the order Avogadro's number or 10^{23}. A slight increase in E makes the extra available phase space volume to all these particles thus greatly increasing the number of possible microstates. As a general result, the *density of states* function increases very rapidly with increase in energy. Consequently, $\Omega(E)$ will increase very rapidly with E while $\Omega'(E^0 - E)$ will decrease very rapidly with increase E. The product of these two functions will result in a very sharp maximum.

Finding the location of this maximum, involves differentiating the equation and setting equal to zero. In this situation, we take partial derivatives with respect to E in order to focus on a single factor being changed while considering other factors as constant. It is more convenient to work with the natural logarithm of the probability equation, which yields:

$$\ln P = \ln \Omega(E) + \ln \Omega'(E^0 - E) + \ln C \tag{8}$$

It becomes much more convenient to work with $\ln P$ instead of P itself as the logarithm is a slower varying function of the energy E, involving the functions Ω and Ω' as a simple sum rather than a product. Thus, at the maximum value of P we have:

$$\frac{\partial \ln P}{\partial E} = \frac{1}{P}\frac{\ln P}{\partial E} = 0 \tag{9}$$

Using Eq. (8), we have:

$$\frac{\partial \ln \Omega(E)}{\partial E} + \frac{\partial \ln \Omega'(E')}{\partial E'}(-1) = 0 \tag{10}$$

which may be rewritten as:

$$\beta(E) = \beta(E') \tag{11}$$

where E and E' represent the energies of System 1 and 2 respectively, along with the definition:

$$\beta(E) = \frac{\partial \ln \Omega(E)}{\partial E} \tag{12}$$

and a corresponding definition for System 2. We now introduce a dimensionless parameter T defined as:

$$\frac{1}{T} = \frac{\partial S}{\partial E} \tag{13}$$

and can introduce the definition:

$$S = k \ln \Omega \tag{14}$$

The quantity S is defined as entropy, while T is defined as temperature. The condition of maximum probability $P(E)$ is then equivalent to the condition that the total entropy:

$$S + S' = maximum \tag{15}$$

and this occurs when:

$$T = T' \tag{16}$$

which corresponds to a situation that the two systems have reached an equilibrium.

III. THE EINSTEIN SOLID AND PHONONS

The Einstein solid is a model of a crystalline solid that contains a large number of independent three-dimensional quantum harmonic oscillators of the same frequency. These oscillators do not interact with each other, they are great options for studying systems with a weak interactions and interatomic coupling. On these scales, the discrete vibrational waves are known as phonons, which exist in different modes, which oscillate around their positions at equilibrium. On nanoscale particle, there are a few quantized phonon modes. In an Einstein solid, we can state that in a single phonon mode, all atoms vibrate independently at one certain frequency. Compared to known continuum models like Debye's, this model can help address the fundamentals of lattice vibrations, although simpler.

We can state that the energies of the solid is given by the expression:

$$E_n = \hbar\omega\left(n + \frac{1}{2}\right) \tag{17}$$

The value n represents the excitation state of ab oscillating mass. However, in this scenario, we can also state that n represents number of phonons that exist at a certain frequency. The term $\frac{\hbar\omega}{2}$ represents the zero-point energy, which, despite the temperature, shows the ground state energy that the oscillator continues to possess. The total energy levels are therefore evenly spaced, and one can define a quantum of energy, or phonon energy, as:

$$\varepsilon = \hbar\omega \tag{18}$$

which is smallest amount by which the energy of an oscillator is increased.

If we want treat this system as a microcanonical ensemble, we need to determine the density of states Ω. That is, the number of ways to distribute q quanta of energy among N' oscillators within a lattice.

We can easily visualize this method of distribution as a combinatorics problem. We can also consider the distribution of q balls over N' containers. Each oscillator occupies a different region on this lattice, and we can state that the quanta represent the balls, while the containers can represent the lattice sites or oscillators itself. We can also assume that each lattice site is different, while every ball is identical. Fig. 2 demonstrates the arrangement of several balls over several bins. We can also reasonably state that the amount of arrangements of N objects is equal to $N!$.

FIG. 2. An illustration of balls distributed in containers, with a changed depiction that suits combinatorics calculations

Therefore, the number of possible combinations of q balls over $N' - 1$ dividers between them is equivalent to the expression $(q + N' - 1)!$. However, because each quanta is identical and the dividers are also identical, many of these permutations are indistinguishable. To correct this overcounting, we can divide by the number of permutations of the identical items. In the end, there are $q!$ identical ball arrangements, and $(N' - 1)!$ identical divider arrangements. We can thus give the multiplicity of the system by the equation:

$$\Omega = \frac{(q + N' - 1)!}{q!(N' - 1)!} \tag{19}$$

The entropy of the system, S, is then directly related to this multiplicity by Eq. (15). We can substitute the expression for multiplicity to obtain:

$$S/k = \ln\Omega = \ln\frac{(q + N' - 1)!}{q!(N' - 1)!}. \tag{20}$$

For a macroscopic system, the number of oscillators and energy quanta are extremely large. Consequently, subtracting one from N' has a negligible effect on the overall calculation. We can therefore approximate the expression for entropy to:

$$S/k \approx \ln\frac{(q + N')!}{q!N'!} \tag{21}$$

In order to approach the complexity of the equation, we can utilize a method known as Stirling's Approximation to simplify entropy. The approximation can provide a value of an estimate of the natural logarithm of a factorial, which we can deploy to yield:

$$S/k \approx (q + N')\ln(q + N') - N'\ln N' - q\ln q. \tag{22}$$

With the now simplified expression for entropy, we can attempt to extract a value for the system's temperature. Inverse temperature is defined by the fundamental thermodynamic relationship, which we can relate to our simplification in Eq. (23).

$$\frac{1}{T} = \frac{\partial S}{\partial E} = \frac{\partial S}{\partial q}\frac{dq}{dE} = \frac{1}{\varepsilon}\frac{\partial S}{\partial q} = \frac{k}{\varepsilon}\ln(1 + N'/q) \tag{23}$$

We can state that the total energy of the solid is given by the sum of the ground state energy and the energy of each ball:

$$E = \frac{N'\varepsilon}{2} + q\varepsilon \tag{24}$$

IV. DEMON ALGORITHM

Computer simulations provide a vehicle for understanding many of the key concepts in statistical mechanics. Monte Carlo simulations are one set of methods to perform these simulations, where a variable is sampled according to the appropriate probability distribution, and averages are computed from these configurations. The demon algorithm is one example of a Monte Carlo method that utilizes results obtained from the microcanonical ensemble.

In this method we slightly relax the total energy constraint. We add an extra particle or degree of freedom to the system. This element is called a *demon*. Since the *demon* consists of just one particle in a system with $N >> 1$, it can be thought of as a very small perturbation to the system. The role of the demon then is to act as a reservoir of energy as it attempts to change the state of the system. The total energy $E = E_s + E_d$ remains unchanged. The demon initially has $Ed = 0$. It then hops from particle to particle, and tries to change the particle's energy. If its desired change lowers the energy of the system, the excess energy is given to the demon. If the desired change raises the energy of the system, the demon gives the required energy to the system, provided that it has enough energy to give. If not, it moves on and tries again with the next particle. The only constraint on this system is that the demon cannot have negative energy.

The entire process is repeated N times, which is referred to as one Monte Carlo sweep. After a sufficient number of such sweeps, the demon and the system will agree upon their average energies. One can associate this average energy to the temperature of the system as will be shown in this section. Thus the demon plays the role of a thermometer.

As we are working with the microcanonical ensemble where all states with some energy E are equally likely, the probability that the demon has some energy E_d is given by

$$P(E_d) \propto \Omega_s(E - E_d) \times \Omega_d(Ed) \tag{25}$$

where Ω_s represents the *density of states* of the system, and Ω_d that of the demon. Since the demon is only one particle (or degree of freedom), it has only one state for each E_d , and so $\Omega_d(Ed) = 1$. This may be rewritten as:

$$\Omega_s(E - E_d) = \exp\left(\frac{S(E - E_d)}{k}\right) \tag{26}$$

and using the definition of temperature, we have:

$$\frac{1}{kT} = \left(\frac{\partial S}{\partial E}\right) \tag{27}$$

If we make a first order Taylor expansion of $S(E - E_d)$, we get:

$$S(E - Ed) \approx S(E) - \left(\frac{\partial S}{\partial E}\right) E_d = S(E) - E_d/kT \tag{28}$$

We can substitute this expansion back into the expression for the density of states and the probability, and can thus write:

$$P(E_d) \propto \exp(-E_d/kT) \tag{29}$$

A. Average Demon Energy and Temperature:Ideal Gas

Let us consider a closed system in which a demon is added, where we adhere closely to a microcanonical ensemble. We can state that there exist N particles of an ideal gas within this system. The Kinetic Molecular Theory allows the assumption that ideal gas particles are void of interactions between each other and have negligible volume. As such, we can assume that all particles have the same amount of velocity.

$$E = \sum_{i=0}^{N} \frac{1}{2} m v_i^2 \equiv \frac{1}{2} \sum_{i=0}^{N} v_i^2, \tag{30}$$

The demon algorithm allows us to sample various microstates in this ideal gas system, performing energy exchanges by slightly changing the velocity of each particle. However, the amount of energy that the demon holds is still significantly smaller than that of the entire system itself. We can initially begin with a one-dimensional model of a system, where we can set the number of particles, N, to 100. We can initially set the total energy of the system, R, to 1000. A random energy exchange is then proposed, an integer value ranging from 0 to 50. This value is assigned with the random action of the demon, giving or taking energy, for initially 100,000 sweeps. Once this system reaches equilibrium, we can analyze the distribution of particle energy. As the system gradually reaches equilibrium, we can utilize temperature to demonstrate the energy being shared among particles. We can also assume that the mass of each particle remains one to ensure simplicity in calculation. To correlate the average demon energy of the system to temperature, we can start by normalizing Eq. (30) utilizing the partition function in order to make sure total energy continues to remain one hundred percent. We can find the normalization constant, which we can set as Z, to be

$$\int_0^\infty P(E_d)dE_d = 1 \tag{31}$$

As such, we can define and solve for the normalization constant Z.

$$P(E_d) = \frac{1}{Z}e^{-E_d/kT} \tag{32}$$

$$Z = \int_0^\infty e^{-E_d/kT}dE_d \tag{33}$$

Solving the integral in Eq. (34) produces the result kT. As the relationship $Z = kT$ is now established, the coefficient $1/Z$ in Eq (33) can be replaced by $1/kT$. We can then redefine Eq (32) as new normalized probability distribution:

$$P(E_d) = \frac{1}{kT}e^{-E_d/kT} \tag{34}$$

This normalized probability distribution can be utilized in the standard definition of an average. We can define the average demon energy as another integral expression:

$$\langle E_d \rangle = \int_0^\infty E_d \cdot P(E_d)dE_d \tag{35}$$

By incorporating Eq. (35) into the expression of P(Ed) into Eq. (36), the integral can be solved to produce the following relationship:

$$\langle E_d \rangle = kT. \tag{36}$$

With the newly derived relationship between average demon energy and temperature, we can verify the validity of the simulation by extracting T from Eq. (30) to numerically visualize energy dispersal after equilibrium. We can extract a value of temperature by linearizing Eq. (30) to plot a line function and extract the slope of the line:

$$\ln P(E) = -E_d/KT + \text{constant} \tag{37}$$

We can set the value for the slope of the line as the inverse of temperature, which we can define as β.

$$\beta = -1/kT \tag{38}$$

We can also report a goodness value for the linear fit using a chi-square analysis. The chi-square value is calculated as the sum of the squared differences between observed and expected frequencies, which each are weighted with expected frequencies. In this case, this can report such a value for Eq. (31). However, regarding a linear fit, we can utilize a method that out puts an R-squared value. This R-Squared value is a metric that ranges from zero to one, and demonstrates the variance proportion from the dependent variable, in our case the natural log of probability density, to the independent variable, in our case the demon energy. We can calculate this R-squared value is calculated by comparing the sum of the squared differences between the data points and the line of regression, alongside the sum of all squared differences from the data average. The measurement of distance from the data points to the line indicates a lower first number, which outputs an R-squared value near one. This value, if near to one, demonstrates a more accurate fit, as points are very close to the line of regression.

The demon algorithm can be adjusted to fit the conventions of the ideal gas simulation. To ensure that the system represented reaches equilibrium, we can graph the number of iterations of Monte Carlo sweeps to the average energy of particles over time to the same conditions established above.

As the demon algorithm exists as a statistical simulation, these properties can serve as a direct analogy to the behavior of real physical systems. We can relate particles in this ideal gas simulation to actual ideal gas particles, following the conventions of the Kinetic Molecular theory. We can relate the behavior of these particles such as Nitrogen gas particles, N_2. However, in moving more realistic renditions of the demon algorithm, it becomes difficult to adhere to the same ideal gas properties. For an Einstein solid model, we can consider alternative methods to better simulate more realistic system.

B. Average Demon Energy and Temperature:Einstein Solid

The demon algorithm can also be adapted to an Einstein Solid model. In order for the model to better fit the conditions, particles can instead be established with varying amounts of energy, set at random. These randomly initialized particles are then randomly picked by the demon, which continues to perform energy exchanges to complete a set number of Monte Carlo sweeps. Following all other initial conditions outlined in the ideal gas simulation, with N set to 100, a total energy of 1000, and an integer delta value between 0 and 50 being selected by the demon as the amount of energy to give or take from a particle at random. The algorithm is set for 100,000 sweeps initially. However, the property defined by Eq. (37) cannot be continuously assumed, as Einstein solids display discrete energy values contrary to the continuous energy found in an ideal gas system. In this system, the value for N can be applied as energy levels occupied at various lattice sites, each contains a set number of energy. In this initial one-dimensional lattice, we can assume that each lattice site starts off with an equal amount of energy. In order to validate the accuracy of the simulation, temperature must be accumulated in alternative ways. The linearized equation Eq. (38) can still be applied to relate the value β to temperature, but as the average demon energy cannot be set to temperature, we can begin by reintroducing Eq. (20) on the multiplicity of the system. This value of temperature, however, must be compared alongside other methods in order to validate the entire simulation. To fit the constraints of a discrete energy system, we can utilize summation instead of the continuous energy assumptions brought by integration to define average demon energy:

$$\langle E \rangle = \frac{\sum_{n=0}^{N} E_n e^{-\beta E_n}}{\sum_{n=0}^{N} e^{-\beta E_n}} \tag{39}$$

By rewriting the numerator as a derivative in terms of β, the equation can be rewritten in a way that agrees with a finite geometric series:

$$\frac{d}{d\beta}\left(\sum_{e} e^{-\beta \cdot \text{in}}\right) \Big/ \left(\sum_{e} e^{-\beta \cdot \text{in}}\right) \tag{40}$$

For a single quantum harmonic oscillator, energy levels are quantized. The partition function for a single oscillator is given by the sum over all these energy states, weighted by their Boltzmann factors, which is given by the common expression in both the numerator and denominator of Eq. (41). By treating the expression as two geometric series in the numerator and denominator, we can simply the equation to produce:

$$\langle E_d \rangle = (1 - e^{-\beta}) \frac{d}{d\beta}\left(\frac{1}{1 - e^{-\beta}}\right) \tag{41}$$

For an Einstein Solid composed of N independent oscillators, the total partition function is simply the product of individual oscillator partition functions. We can apply this to the Einstein Solid's partition function to obtain an expression for the average energy per oscillator. If we consider the average energy held by the demon as the system's average energy per degree of freedom, we can produce a final expression for average demon energy to be:

$$\langle E_d \rangle = \frac{1}{e^{\beta} - 1} \tag{42}$$

As $\langle E_d \rangle$ does not directly define temperature in a discrete energy system, β must be extracted. Immediately, we can attempt to rearrange this expression to solve for β. We can rearrange Eq. (43) to isolate the term with β, and then use a natural logarithm to generate a valid expression for β:

$$\beta = \ln\left(1 + \frac{1}{\langle E_d \rangle}\right) \tag{43}$$

Upon cross-verification of the algorithm, we can confirm that all values of temperature arrive at the same value. As a viable next step for adapting the demon algorithm to a more realistic scenario, the scenario can be upgraded to a two-dimensional lattice, contrary to the one-dimensional ten-by-ten grid simulated currently. In this more realistic lattice, the demon travels between lattices by moving in four different directions: up, down, left, and right. Upon reaching the edge, the demon wraps around to the other side and continues. However, the realistic attributes of the current random pick case become increasingly less accurate. Compared to an ideal gas, the application of a random walk algorithm is more preferably utilized in an Einstein Solid due to the lattice arrangement. In an ideal gas, as the particles are all non-interacting without fixed positions, they can freely move in the container. As such, the random pick model, where any particle can be chosen for an energy exchange, is much more fitting for an ideal gas It is assumed that any particle can exchange energy with any other, regardless of their spatial position and distance. In a true crystalline solid, however, the propagation of energy is a process that remains localized, through nearest-neighbor interactions between particles. The random pick algorithm is therefore more unrealistic in this case, completely lacking these nearest-neighbor interactions. As such, a random walk algorithm can be implemented. In a random walk case, the demon seemingly paces around the lattice, making a decision on which direction to move from its current location. The demon, therefore, moves from one lattice site to a neighboring lattice site, similarly to actual diffusion.

In real dissipation of heat, however, this process is not as simple as a coin toss, but rather aligning with a certain probability weightage between each decision. Let the current position of the demon be n_0, and let i represent a random integer between 1 and 4, each denoting a different direction the demon will move. The probability of the demon moving to a neighboring site, i, can be denoted by the expression:

$$P_i = \frac{|n_o - n_i|}{\sum_i |n_o - n_i|} \tag{44}$$

Upon the demon reaching a new site, the probabilities of the demon either taking or giving energy are highlighted by a different probability weightage, shown respectively as taking and giving in Eq. (48) and Eq. (49)

$$\frac{n_0}{n_0 + n_i} \tag{45}$$

$$\frac{n_i}{n_0 + n_i} \tag{46}$$

These weightings can be incorporated into the random walk algorithm to better suit realistic heat dissipation in the lattice. These probabilities, which depend on the energy states of interacting sites, align with the second law of thermodynamics, which favors, but is not exclusive to, the transfer of energy from hotter to colder regions.

V. RESULTS

A. Ideal Gas Demon Algorithm

Upon running the demon algorithm for an ideal gas with 100 particles, a total energy of 1000, and energy exchanges with integer values ranging from 0 to 50, we can determine if the system successfully reached equilibrium. By graphing each iteration of Monte Carlo sweep to the corresponding energy of each particle we can see the average energy slowly stabilizing over time. We can also graph the number of iterations to Demon Energy, and see the values stabilizing as well, except at a lower value.

Fig. 3 presents two diagnostic plots that demonstrate the behavior of the demon algorithm over the course of the simulation. As the demon gradually performs more energy transactions with each particle during a Monte Carlo Sweep, energy spreads out between particles until the system approaches a thermal equilibrium. In Fig. 3a, the average energy per particle as a function of time stabilizes as the simulation progresses. This plateau indicates that the system has reached equilibrium, with energy uniformly redistributed among particles. The average demon energy reaches a steady value at equilibrium in Fig. 3b, the opposite pattern of the average energy per particle. Over several iterations, the energy of the demon stops changing dramatically as the average individual particle energy settles to an equilibrium. We can apply an exponential fit to the exponentially decaying demon energy, adhering to the relationship in Eq. (30). The fit, following a theoretical Boltzmann Distribution, can be compared with the average demon energy which is binned and plotted as histograms.

The histogram of the demon's energy counts the number of occurrences of varied energy values. The exponential decay of average demon energy is best visualized in Fig. 4, as the system approaches equilibrium. The overlaid red

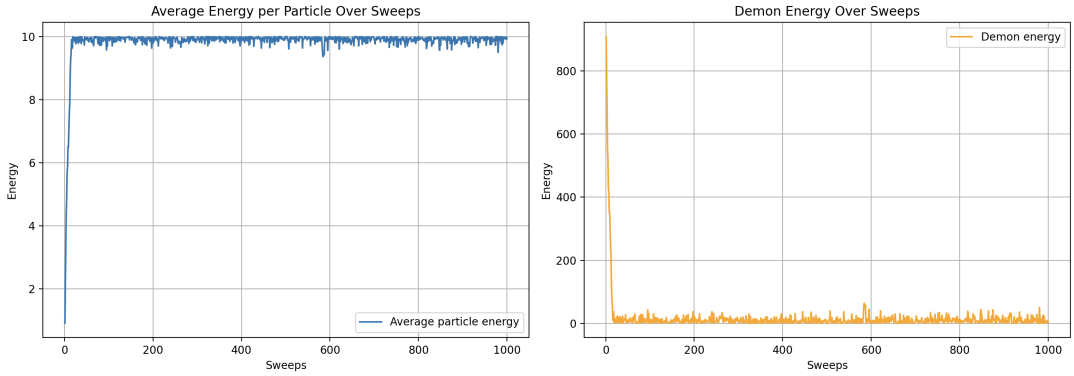

FIG. 3. Simulation of ideal gas particles over iterations of Monte Carlo sweeps. Fig. 3a represents the average energy of each particle over iterations, while Fig. 3b demonstrates the exponentially decaying relationship of average demon energy over successive sweeps.

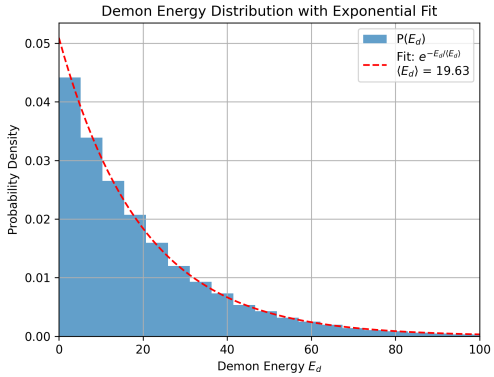

FIG. 4. Exponential fit applied to histogram plot of demon energy to probability density.

dashed line in Fig. 4 represents the non-linear exponential fit applied onto the empirical histogram data. This fit yields a characteristic average demon energy of 19.63, which is outlined in the legend. The agreement between the empirically obtained probability distribution and the theoretical exponential fit illustrates the system successfully reaching thermal equilibrium. This visualization directly supports Eq. (30), stating that the likelihood of observing the demon with a given amount of energy decays exponentially as energy increases To further validate this exponential relationship and to extract the inverse temperature β from the distribution, we can plot the linearized probability function given by Eq. (38). We can plot the natural logarithm of the probability density against the demon's energy. According to the Boltzmann distribution, such a plot should yield a linear relationship with a slope equal to negative β. We can extract the value β from the plot and calculate the temperature accordingly. Fig. 4 presents this direct linear transformation. As predicted by the Boltzmann distribution, the blue data points representing the natural log of probability density exhibit a linear relationship with increasing demon energy. The red dashed line is a linear fit applied to this transformed data. Seen in Eq. (39), the slope of the plot of energy probability density is equal to the negative of the inverse temperature. The slope of this linear fit is determined to be -0.05, as indicated in the legend. From the exponential fit in Fig. 3, the value of average demon energy is once again given as 19.63. Using the slope value of 0.05 from Fig. 5 and applying it to Eq. (39) as β. produces a value for temperature to be 20. The close agreement between these two independently determined values of temperature affirms the validity of the Monte Carlo simulation and confirms that the system has successfully reached a thermal equilibrium, with a percent discrepancy value of 1.85 %. We can also report an R-squared value to indicate the goodness of the linear fit, which is yielded to be 0.9923, indicating an excellent fit.

Subsequently, we can translate the dimensionless temperature units into a more physical context, assuming nitrogen dioxide particles as the particles within the system. With the demon algorithm exhibiting success in the simulation with an ideal gas, it can be refined to further simulate alternative systems which we will be subsequently explore.

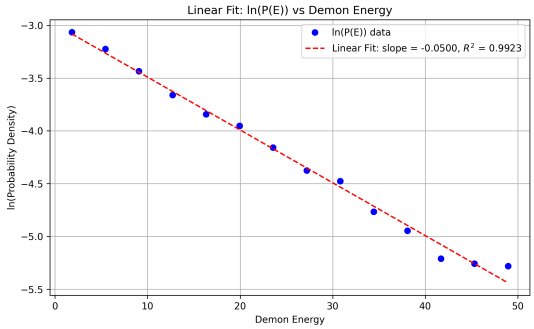

FIG. 5. Linear fit applied to demon energy plotted against the natural logarithm of probability density to obtain a linearized plot. The plot is cut off at a hundred to restrict additional noise after equilibrium. The Linear fit slope is shown to be -0.05, outlined in the legend.

B. Einstein Solid Demon Algorithm

We can furthermore run the adapted Demon Algorithm for the Einstein Solid.

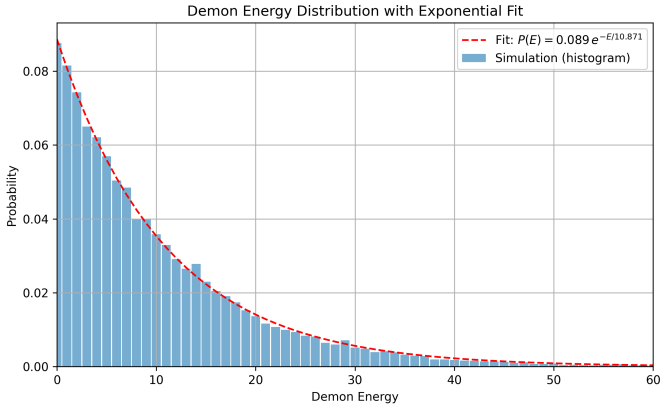

FIG. 6. Exponential fit applied to a plot of the demon algorithm adapted for an Einstein solid graphing demon energy against probability density. The average demon energy is given by 9.720, through a terminal output. The average is calculated by finding the arithmetic mean of the demon's energy across all Monte Carlo sweeps.

By simulating the demon algorithm through a software program, the average demon energy was computed to 9.720. The average is calculated by finding the arithmetic mean of the demon's energy across all Monte Carlo sweeps. Despite fitting a discrete energy system, demon energy continues to demonstrate the same exponential decay relationship. In Fig. 6, the histogram bars represent the empirical frequency of observing demon energy at different energy levels. As highlighted again by the Boltzmann distribution, the probability of the demon possessing higher energy states diminishes exponentially, similar to the ideal gas rendition in Fig. 4. A linear fit can be reapplied to redetermine the new value for β to extract temperature using Eq. (39).

The linear plot in Fig. 7 presents a slope value of -0.097, which using Eq. (39) produces the value of 10.3092. The plot also subsequently produced an R-Squared value of 0.9965, indicating a strong linear fit.

Using the rearranged function given by Eq. (44), the temperature is given to be 10.22084803, which is decently agreeable to the value of temperature given through the linear plot in Fig. 7, with a percent discrepancy of 0.86%. Finally, the value for temperature can be obtained through one more way, utilizing Sterling's Approximation from Eq. (23). From this simplified entropy expression, the temperature can be derived through its fundamental thermodynamic definition described in Eq. (14). For the Einstein Solid, where total energy is defined by Eq. (25), assuming zero-point energy is absorbed, this derivative simplifies to the expression presented in Equation (24). An algorithm can be implemented and set to the same conditions as the demon algorithm, with a system where $N = 100$ and $R = 1000$. The said algorithm yields a temperature of approximately 10.589 energy units, agreeing with the Fig. 7 derivation with a percent discrepancy of 2.76% and agreeing with the algebraic derivation with a percent discrepancy of 3.63%.

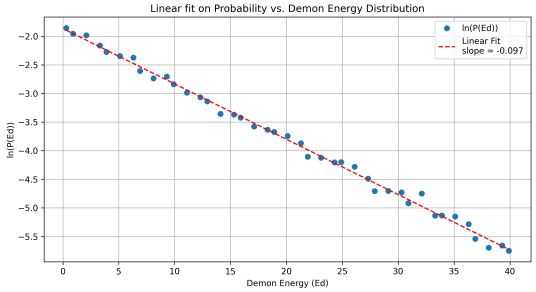

FIG. 7. Linear fit applied to plot of demon energy against the natural log of probability density. The plot is cut as the demon energy reaches forty to restrict additional noise after the system reaches equilibrium. The slope value is given to be -0.097, shown in the legend.

With relatively agreeable success between the three values of temperature, we can assume that the demon algorithm adapted for an Einstein solid is valid.

Upon implementing a Random Walk algorithm with the same number of particles, total energy, and energy exchange changes, we can once again plot a linear fit to determine the temperature of the system. For a two-dimensional lattice, we can simulate results initially on a 10x10 grid to match with results from the random pick scenario.

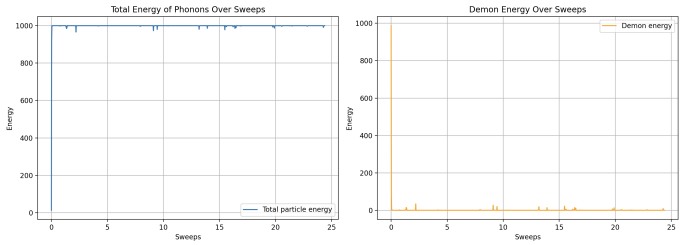

FIG. 8. Simulation of particles over iterations of Monte Carlo Sweeps. Figure 8a represents the average energy of each particle, while Figure 8b demonstrates the exponentially decaying relationship of average demon energy over successive sweeps.

The new Random Walk algorithm also exhibits a similar trend in the decay of stabilization of both particle and demon energy over time, seen in Fig. 8a and Fig, 8b respectively. The discrete energy compared to continuous energy is more easily visualized in these plots, with a only discrete integer values serving as the quantized energy levels. As the probability of having a particle at a higher energy is so substantially small, a small percent of particles exist at such a level. We can also test a larger sample set of ten thousand particles to ensure the movement of the system to equilibrium aligns with the Boltzmann Distribution we have continued to see. In Fig 9, we see the distribution of total energy graphed against probability distribution as a function of decaying energy, aligning with a Boltzmann Distribution. This confirms that the system has reached thermal equilibrium.

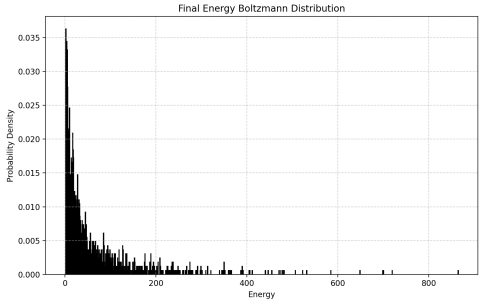

FIG. 9. The distribution of demon energy to probability density of ten thousand particles implemented in a Random Walk algorithm

We can also plot a linear fit to the distribution to obtain a value for temperature for the purpose of algorithm validation. The graph of a linear fit in Fig. 10 produces a slope of -0.0913. Using Eq. (39) returns a temperature of

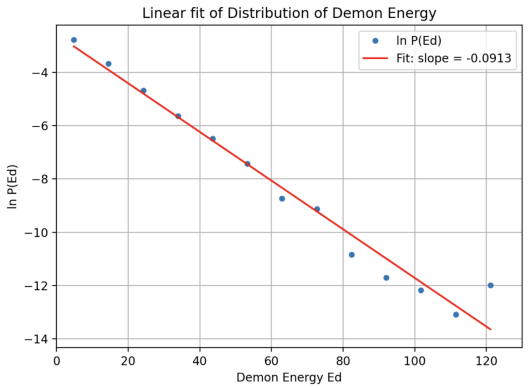

FIG. 10. Results of a linear fit of demon energy to the natural logarithm of probability density in a 10x10 lattice demon algorithm with an implemented random walk. Results are restricted to 130 to prevent additional noise and the slope is given as -0.0913 as seen in the legend.

10.9529, with a percent discrepancy between the random pick linear fit in Fig. 6 of 6.05%. The linear fit also produces an R-squared value of 0.9976, which again positively aligns with the accuracy of the fit. With a decent agreement of temperature results with the random pick scenario, the random walk algorithm can be updated to fit a 64x64 lattice. However, currently, the decision of a demon to move from its current location to its next location is governed by a random choice between two integers, -1 and 1.

In a 64x64 lattice, we can also determine the pathway of the demon as it paces around the lattice from neighbor to neighbor. With the probability weightage set by Eq. (48) and Eq. (49), Fig. 11 describes a visualization of the demon's path.

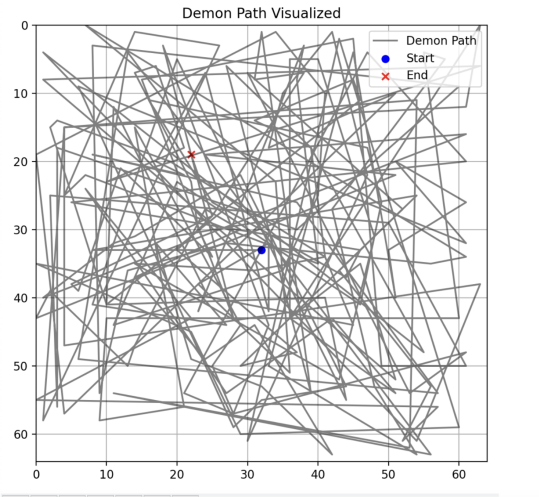

FIG. 11. Visualization of the demon's path through a 64x64 lattice. Decisions of moving, giving, and taking energy are all made through probabilistic weightages.

VI. CONCLUSION

In this study, we have explored the demon algorithm as an efficient computational tool for simulating thermal equilibrium in both ideal gases and the Einstein Solid. Through a series of cross-references to ensure algorithm validity, we have demonstrated the effectiveness in accurately simulating both scenarios.

The progression of plots demonstrated the systematic construction of a suitable demon algorithm to fit an Einstein

solid, with changes being made to reach more realistic depictions of heat dissipation. The calculations and results demonstrate the demon algorithm's gradual approach of the system towards thermal equilibrium. Aligning with the Boltzmann distribution, demon energy consistently exhibits an exponential decay. This agreement highlights that the system successfully reached equilibrium, with an obtained value for temperature. However, compared to a typical time metric, the movement of the system towards equilibrium was tracked through successive Monte Carlo sweeps, iterating through energy exchanges between the demon and particles until equilibrium was successfully reached, seen in Fig. 3.

With a more realistic model of the Einstein Solid, the algorithm can be run to simulate the dissipation of heat in a system, which can serve benefits in devices on quantum scales. With discrete energy governing systems of that scale, the effective management of heat transfer becomes critical to track to prevent issues such as overheating. The distribution of heat is highly critical with systems to prevent the breakdown of small components, to ensure complete productivity and efficiency of such nanodevices. Thermal transport through phonon scattering can be seen in components such as nanowires or thin films can be sampled through the demon algorithm, with varied input parameters and adjustments to fit any constraints. The model can continue to be pushed towards a more realistic scenario, such as varied lattice shapes or varied potentials. By fitting to constraints of more complicated crystals, such as Gallium-Arsenide crystals, more constraints of heat can be incorporated to apply to more complex scenarios. In all, this study validates the potential of the demon algorithm as an efficient method for simulating thermal equilibrium adapted to different systems, which can be enhanced to address more complex problems in the future.

VII. REFERENCES

[1] Kurt Binder and Dieter W. Heermann. *Monte Carlo Simulation in Statistical Physics*. Springer-Verlag, 1988.

[2] J. Bonde and R. Haukland. On the combinatorics of placing balls into ordered bins. *Acta Arithmetica*, 153:197–208, 2012.

[3] Philip Cherian. Computational assignment the demon algorithm. *Monsoon*.

[4] Michael Creutz. Microcanonical monte carlo simulation. *Physical Review Letters*, 50(19):1411, 1983.

[5] Frederick Reif. *Fundamentals Of Statistical and Thermal Physics*. McGraw-Hill, 1965.

[6] Jan Tobochnik and Harvey Gould. *Understanding the temperature and the chemical potential using computer simulations*. 2006.

[7] Jan Tobochnik, Harvey Gould, and Wolfgang Christian. *Monte Carlo Simulations of Thermal Systems*. 2006.