

Simulating an Ideal Gas to Verify Statistical Mechanics

Jeffrey Chang

Department of Physics, Stanford University

(Dated: June 3, 2018)

I simulate an ideal gas and observe how its macroscopic properties emerge from microscopic time evolution. First, I implement an event-based molecular dynamics engine of hard-sphere particles bouncing around inside a box. With this engine, I demonstrate that (1) particle speeds approach a Maxwell-Boltzmann distribution, (2) that the system agrees with the ideal gas law $PV = Nk_B T$, and (3) that fluctuations in local density decrease with N . These simulations verify that statistical mechanics is indeed a reliable framework for modeling large numbers of particles.

I. INTRODUCTION

How do macroscopic "thermodynamic" properties of matter – temperature, pressure, and more – emerge from the microscopic laws of physics? Statistical mechanics provides a simple and elegant framework to answer this question. It explains how thermodynamic properties result from the dynamics of many small particles. However, the main concepts and assumptions of statistical mechanics are rather abstract and intangible, and furthermore, they are not often "proven" in the curriculum in an intuitively convincing way. For these reasons, it is instructive to explicitly visualize how a large system attains its thermodynamic properties.

Here, I simulate an ideal gas, particle-by-particle, to demonstrate that its large-scale properties indeed result from the dynamics of individual particles. In particular, I demonstrate that (1) non-equilibrium states evolve to equilibrium; (2) the gas satisfies the equation of state $PV = Nk_B T$; and (3) fluctuations around equilibrium occur, and decrease with large N . Afterwards, I briefly discuss the ergodic assumption, and then I show that the simulation can account for non-equilibrium phenomena such as Brownian motion.

II. DESCRIPTION OF IDEAL GAS

In this section I will describe the ideal gas microscopically, and in later sections, I will explain the emergent properties of entropy and pressure as they naturally arise.

The 'ideal gas' is a simple statistical mechanical model that successfully explains many properties of real gases such as their heat capacities. I choose to simulate an ideal gas because it is the simplest model that demonstrates main tenets of statistical mechanics.

My simulation consists of N classical particles with positions \vec{x}_i and velocities \vec{v}_i , contained in a box of side length L . Each particle is a "hard-sphere" of radius r_i and mass m_i . The particles exert no forces on each other, but they collide elastically with each other and reflect (specularly) on the walls of their container. We can consider these collisions as delta-function force spikes that instantaneously change the particle velocity. Note that since all interactions are elastic, the total energy of the

system

$$E = \sum_i \frac{1}{2} m_i |\vec{v}_i|^2 \quad (1)$$

is conserved.

A. Equations of motion

Now we describe the time evolution of these colliding classical particles. When a particle is traveling freely through space, it moves at a fixed velocity with a trajectory given by

$$\vec{x}_i(t + \Delta t) = \vec{x}_i(t) + \vec{v}_i \Delta t. \quad (2)$$

A particle reflects off a wall with unit normal \hat{n} at position w along the \hat{n} coordinate if the particle touches the wall; that is, if

$$\vec{x}_i - r_i \hat{n} = w. \quad (3)$$

Since we only consider specular reflection, the component of velocity normal to the wall is inverted, meaning that its velocity changes as

$$\vec{v}_i \rightarrow \vec{v}_i - 2(\vec{v}_i \cdot \hat{n})\hat{n}. \quad (4)$$

A collision between two particles i and j occurs when

$$|\vec{x}_i - \vec{x}_j| = r_i + r_j, \quad (5)$$

or when the particles touch. During the collision, a momentum \vec{q} is exchanged between the particles:

$$\begin{cases} \vec{v}_i \rightarrow \vec{v}_i + \vec{q}/m_i, \\ \vec{v}_j \rightarrow \vec{v}_j - \vec{q}/m_j. \end{cases} \quad (6)$$

To fix \vec{q} , we apply two constraints. Since we only consider elastic collisions where kinetic energy is conserved, we must satisfy

$$\frac{1}{2} m_i v_i^2 + \frac{1}{2} m_j v_j^2 = \frac{1}{2} m_i |\vec{v}_i + \vec{q}/m_i|^2 + \frac{1}{2} m_j |\vec{v}_j - \vec{q}/m_j|^2, \quad (7)$$

and since we only consider “specular collision”, \vec{q} must lie normal to the reflection plane; that is,

$$\vec{q} = q \hat{r}_{ij}, \quad (8)$$

where $\hat{r}_{ij} \equiv (\vec{r}_i - \vec{r}_j)/|\vec{r}_i - \vec{r}_j|$ is a unit vector pointing between the centers of the particles. Solving these equations for q gives us

$$\vec{q} = -2 \frac{m_i m_j}{m_i + m_j} [(\vec{v}_i - \vec{v}_j) \cdot \hat{r}_{ij}] \hat{r}_{ij} \quad (9)$$

as the momentum transferred when two particles collide.

Given these equations of motion, the task of our simulation engine is threefold: (i) to propagate the particles through space with Eq. 2 when they are not colliding; (ii) to update the velocities with Eq. 4 whenever the particles collide with the walls (Eq. 3); and (iii) to update the velocities with Eq. 6 whenever the particles collide with each other (Eq. 5).

III. IMPLEMENTATION

Now I describe my tactics for simulating the physics described in section II A. In brief, I implement an event-based molecular dynamics engine in python, using the numpy and matplotlib libraries.

A. Description of event-based molecular dynamics

Here I outline the algorithm of event-based molecular dynamics. I learned most of this information from reference [1].

The main idea of event-based molecular dynamics is to proceed collision by collision, rather than timestep by timestep. To do this, the engine maintains a `EventQueue` of upcoming collisions, sorted by their time of occurrence. At each step, the engine determines pops off the next collision from the `EventQueue`, evolves time until this collision using Eq. 2, and then performs this collision using Eq. 4 or Eq. 6.

To maintain the correctness of the `EventQueue`, the engine needs to update the timings of future events whenever a particle involved in that event collides. In other words, each time a particle p undergoes a collision, the engine needs to re-calculate the times of all future events involving particle p , and update the `EventQueue` accordingly. The timings of future events can be determined by solving for t in Eqs. 3 and 5.

The data structure used to maintain the `EventQueue` is a priority queue, which contains many ‘events’, each associated with a ‘priority’. A priority queue can extract the highest-priority event in $O(\log N)$, and can update the priority of existing events in $O(\log N)$. In my case, the event is a pair of colliding particles, and the priority

is the time of collision. The priority queue data structure allows me to extract the next collision in $O(\log N)$ time, and allows me to update the time of upcoming collisions in $O(\log N)$ time.

The pseudocode for the algorithm is shown below.

```

***

[Initialize particle positions and velocities]

/* Initialize the Event Queue -- O(N^2)*/
queue <- [New Event Queue]
for [all pairs of particles]:
    t <- [Calculate time until particles collide]
    queue.push(t, particle pair)

[Sort the event queue by time]

While true:
    /* Retrieve next event from the Queue
    -- O(log N) */
    t, particles <- queue.pop()

    /* Evolve all particles until this time
    -- O(N) */
    for [all particles]:
        particle.evolve_until(t)

    /* Perform this event -- O(1) */
    particles.collide()

    /* Update the timings of all particles
    involved in this event -- O(N log N)*/
    for [particle in this event]:
        for [upcoming events
            involving this particle]
            time <- [Get new collision time]

        event.update_priority(time) // O(log N)

***

```

B. Software Architecture

I choose a clean and modular software design. A diagram of the architecture can be found at the end in Figure 6.

C. Verification

I implemented and verified my event-based molecular dynamics engine as described in my progress reports. The individual components of the system were designed and verified separately. For instance, when I implemented the particle collision velocity update rule (Eq. 6), I tested the function extensively with various initial particle velocities and positions, and I checked that the

new velocities were physical and that energy was conserved.

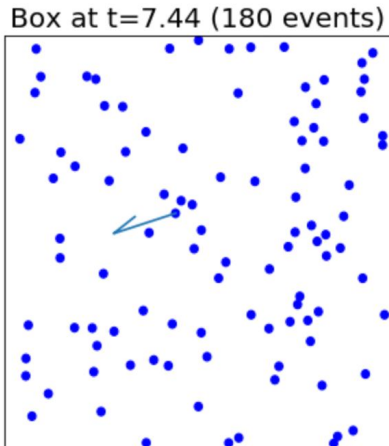


FIG. 1. Shown above is a frame of an animation produced by my event-based molecular dynamics engine. The behavior of the particles appears physically probable. In light blue is the path of one of the particles traced over time.

Once I implemented all the components of the engine, I manually verified the overall correctness by drawing the particles and printing the `EventQueue` after each collision. The resulting animations appeared visually correct (Fig. 1). As one final sanity check, the overall system energy is conserved as particles collide (data not shown).

With this molecular dynamics engine in place, I am ready to simulate an ideal gas and observe its emergent features.

IV. RESULTS

I find that this simulation can reproduce many of the known and predicted properties of ideal gases. Below, I conduct a total of 3 experiments with ideal gases, demonstrating the Maxwell-Boltzmann distribution, the ideal gas law $PV = Nk_B T$, and fluctuations around equilibrium. For each of these properties, I will first show the experimental result, and then give the theoretical justification.

Animations are more lucid than the static figures in this pdf. The gifs used in my presentation can be found under <http://stanford.edu/~jeffjar/files/physics113/>.

A. Maxwell-Boltzmann distribution: Experiment

First I observe the speed distribution of the particles evolves in time. I find that it approaches a definite shape

rather quickly (in roughly $O(N)$ collisions) and then stays close to this equilibrium shape.

Surprisingly, no matter how the particles are initialized, the distribution of their speeds approaches the same shape. Shown in Figure 2 is how the speed distribution of $N = 100$ particles with total energy $E \approx 2000$ changes in time. A initial uniform speed distribution evolves to an equilibrium distribution and stays there (top panel). Even if I initialize the speeds in a very non-equilibrium manner, where one particle carries all 2000 units of energy, and all the other particles are at rest, the distribution of speeds approaches equilibrium within 500 collisions (middle panel). When the speeds are initialized with this distribution, they stay near it (bottom panel).

The experimental tendency of the particles to approach a certain “equilibrium” of speeds begs for a simple theoretical explanation. Looking at the animations under <http://stanford.edu/~jeffjar/files/physics113/>, I can make some suggestive observations: there are many possible ways for the speeds to distribute themselves in the equilibrium manner (Fig. 2, right panel), but not so many ways for the initial non-equilibrium manner. This observation suggests a statistical approach of counting the “ways” or “complexions” of comprising different distributions.

Below, I will explain the equilibrium distribution of speeds using statistical mechanics.

B. Maxwell-Boltzmann distribution: Theory

Here, I will go through classic argument of maximizing entropy to derive the Maxwell-Boltzmann speed distribution. To begin, I will use the maximum-entropy argument to derive the Boltzmann factor.

Consider a system of N particles sharing a total energy E . Suppose that the possible energies are given by ϵ_i , and that there are n_i particles having energy ϵ_i . We wish to know the “equilibrium” distribution of energies – the $\{n_i\}$, telling us how many particles have energy ϵ_i .

As suggested earlier, the system will evolve towards $\{n_i\}$ s with a greater number of “complexions” or “microstates” – different ways of assigning energy to individual particles that still arrive at the same overall distribution. In particular, the number of complexions W is given by the combinatoric expression

$$W = \frac{N!}{\prod_i n_i!}, \quad (10)$$

where the $N!$ in the numerator accounts for the total number of ways of re-assigning energies to particles, and the factors of $n!$ in the denominator correct for over-counting for different particles having the same energy ϵ_i . Since this expression scales exponentially with N , it is nice to take the logarithm of the complexion. We can use the Sterling approximation $\log n! \approx n \log n - n$ to

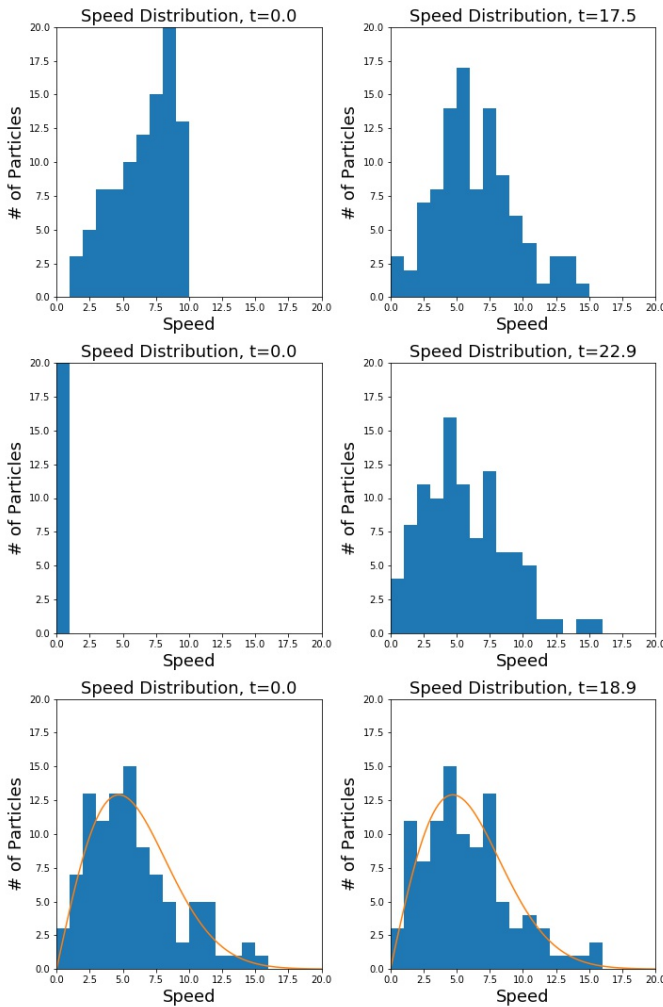


FIG. 2. The speed distribution quickly evolves to the same distribution, regardless of the initial conditions. Left, the speed distribution at $t = 0$; Right, the speed distribution after 500 collisions. Top, uniform initial velocity distribution; Middle, uniformly-zero initial velocities, with one nonzero velocity; Bottom, Maxwell-Boltzmann initial conditions, with the theoretical distribution (Eq. 19) shown in orange. All simulations are in 2D, $V = 100^2$, $N = 100$, $E \approx 2000$.

simplify the expression in limits of large n :

$$\log W \approx N \log N - N - \sum_i (n_i \log n_i - n_i) \quad (11)$$

$$= N \log N - \sum_i n_i \log n_i. \quad (12)$$

We wish to find the distribution $\{n_i\}$ that maximizes $\log W$, subject to the constraints

$$\sum_i n_i = N; \quad \sum_i \epsilon_i n_i = E.$$

Introducing the Lagrange multipliers α and β to imple-

ment these constraints, we find that we must extremize

$$G(\{n_i\}) \equiv \log W + \alpha(\text{constraint 1}) + \beta(\text{constraint 2}) \quad (13)$$

$$= N \log N - \sum_i \log n_i + \alpha(N - \sum_i n_i) + \beta(E - \sum_i \epsilon_i n_i) \quad (14)$$

Taking the derivative of G with respect to n_j , we find

$$\frac{dG}{dn_j} = -\log n_j - 1 - \alpha - \beta \epsilon_i, \quad (15)$$

and setting this to zero and solving for n_j , we arrive at the result

$$n_j = e^{1-\alpha} e^{-\beta \epsilon_i} \implies \boxed{\frac{n_j}{N} = \frac{1}{Z} e^{-\beta \epsilon_i}}, \quad (16)$$

where we define $Z \equiv N e^{1+\alpha} = \sum_i e^{-\beta \epsilon_i}$. This is our desired Boltzmann distribution; it says that the $\{n_i\}$ distribution with the most complexions is a decaying exponential in energy.

To derive the speed distribution from the energy distribution, we need to do a few additional steps. Since the kinetic energy of a particle is given by $\epsilon = \frac{1}{2} m v^2$, the distribution over velocities is given by

$$P(\vec{v}) d\vec{v} \propto e^{-\beta m |\vec{v}|^2 / 2}, \quad (17)$$

a Gaussian in all dimensions of velocity with spread $1/\sqrt{\beta m}$. To arrive at the distribution over speeds, we assume an isotropic distribution of velocities, and average over angles. In three dimensions, the volume element is $d\vec{v} = 4\pi v^2 dv$, so we arrive at the final result

$$\boxed{P(v) dv \propto v^2 e^{-\beta m v^2 / 2}}. \quad (18)$$

In two dimensions, the area element is $d\vec{v} = 2\pi v dv$, so the distribution is instead given by

$$P(v) dv \propto v e^{-\beta m v^2 / 2}. \quad (19)$$

Shown in Figure 2 is a fit of this theoretical distribution over the observed experimental distribution. As seen in the figure, the simulated speeds are explained quite convincingly by the Maxwell-Boltzmann distribution, suggesting that the theoretical framework of maximizing entropy and approaching equilibrium *does* reasonably explain the behavior of many particles.

I have shown that the Maxwell-Boltzmann distribution can explain my observed distribution of speeds. This distribution, however, is parametrized by the Lagrange multiplier β , which I have not yet related to a physical property of my simulation. In the next section I will relate β to E , the total energy of all particles in the simulation. Defining $k_B T \equiv 1/\beta$ as the temperature, the main result is that $E = \frac{d}{2} N k_B T$, where d is the number of dimensions in the simulation.

We will now find the energy of a collection of N particles obeying a Maxwell-Boltzmann distribution of velocities (Eq. 17). In d dimensions, the kinetic energy of a single particle is given by

$$\epsilon = \frac{1}{2}m \sum_{i=0}^d v_i^2, \quad (20)$$

where v_i is the i 'th component of velocity. The total energy of N particles obeying this distribution is

$$E = N\langle\epsilon\rangle \quad (21)$$

$$= \frac{1}{2}Nm \int \sum_{i=0}^d v_i^2 \cdot P(\vec{v})d\vec{v}. \quad (22)$$

Since $P(\vec{v})$ is the same for all components of v , we can just evaluate the sum for one component i , and multiply the result by d . To do this, we first rewrite the velocity distribution component-wise as

$$P(\vec{v})d\vec{v} \propto e^{-\beta m \sum_{j=0}^d v_j^2/2} d\vec{v} \quad (23)$$

$$\propto \prod_{j=0}^d e^{-\beta m v_j^2/2} dv_j. \quad (24)$$

When we integrate over each of the v_j s, each integral yields a factor unity (because the distribution is normalized), except for the $i = j$ integral, which gives us

$$\int_{-\infty}^{+\infty} v^2 e^{-\beta m v^2/2} dv = \frac{1}{2} \sqrt{\pi \left(\frac{2}{\beta m} \right)^3} \quad (25)$$

$$= \sqrt{\frac{2\pi}{(\beta m)^3}}. \quad (26)$$

Putting everything together, including the normalization constant $A = \sqrt{\beta m/2\pi}$, we find that the energy is

$$E = \frac{d}{2}mNA \int v^2 P(v)dv \quad (27)$$

$$= \frac{d}{2}mN \sqrt{\frac{\beta m}{2\pi}} \int_{-\infty}^{+\infty} v^2 e^{-\beta m v^2/2} dv \quad (28)$$

$$= \frac{d}{2}N \frac{1}{\beta}, \quad (29)$$

so indeed,

$$\boxed{E = \frac{d}{2}Nk_B T}. \quad (30)$$

I want to make one last note before moving on. There are d contributions to the kinetic energy; each component of the velocity adds a $m\langle v_i^2 \rangle/2$ to kinetic energy. Each

of these ‘‘translational degrees of freedom’’ in turn contributes an energy of $k_B T/2$. This is an example of the so-called equipartition theorem from classical statistical mechanics.

This result concludes my discussion about the equilibrium distribution of speeds in an ideal gas. Maximizing the entropy leads to a Maxwell-Boltzmann distribution of velocities (Eq. 17), parametrized by the Lagrange multiplier β . As shown in treatments of statistical mechanics [2], this β is the reciprocal of the temperature $k_B T$. In turn, the temperature of an ideal gas is related to its energy by equation 30. I have thus related the equilibrium distribution of speeds of the ideal gas (in the NVE ensemble) to the total energy of the particles.

Now that I have demonstrated explicitly the approach towards equilibrium, I will investigate the equilibrium property of pressure.

C. Ideal Gas Law: Experiment

When the particles collide with the walls, they deliver momentum to the walls. This net momentum delivery results in an average force per area of wall – the pressure of the gas. A natural question to ask is how this pressure varies with other macroscopic variables of the gas, such as the number of particles N , the volume V , or the temperature T (for our purposes, temperature is defined by $E = \frac{d}{2}Nk_B T$).

Shown in Figure 3 is how the simulated pressure P depends on the volume V , the number of particles N , and the temperature $k_B T$. In each of these plots, the only the variable of interest is varied, while the others are held constant. As seen in the plots, the pressure varies inversely with volume and linearly with temperature and number of particles: $P \propto Nk_B T/V$. In fact, looking more carefully we see that the constant of proportionality is unity. The simulated system obeys the relation $PV = Nk_B T$ – the ideal gas law.

D. Ideal Gas Law: Theory

Now I will derive the ideal gas law.

There are a number of ways to derive the ideal gas law. The most general way, based in statistical mechanics, is by taking the appropriate derivatives of the partition function Z and then solving for relationships between the thermodynamic variables. However, these approaches rely on a background framework of statistical mechanics, which I do not have the room to construct. Instead, here, I will take a simpler microscopic approach in the spirit of ‘‘kinetic molecular theory’’. I think this approach lies more in line with the explicit nature of my simulation.

Let us consider the momentum delivered to a wall patch of area A during time τ . The only momentum comes from the component of velocities normal to the

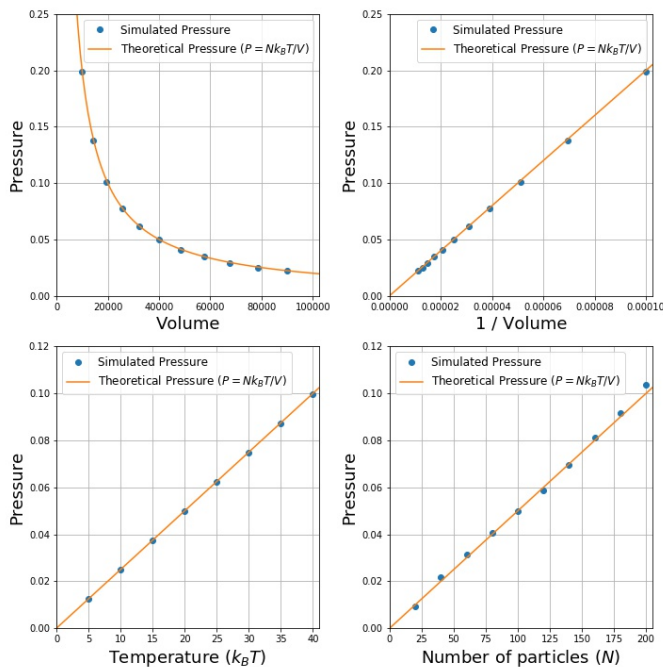


FIG. 3. The pressure of the simulated ideal gas agrees very well with the theoretical $P = Nk_B T/V$ prediction. Clockwise from top left: Pressure to volume, pressure to inverse volume (for linearization), pressure to number of particles, pressure to temperature. All simulations are in 2D, and the non-varying parameters are held at $V_0 = 200$, $k_B T_0 = 20$, $N_0 = 100$. The measured pressure is averaged over 500 events. Particles are initialized with position drawn from a uniform distribution within the boundaries and velocities drawn from a Maxwell-Boltzmann distribution of temperature kT .

wall, v_x . For now, we consider only particles with velocity component v_x ; to arrive at the final result, we integrate over the x -components of velocity, weighted by the Maxwell-Boltzmann distribution.

To determine the momentum transmitted to the wall, we multiply the momentum transferred per collision by the number of collisions. Each collision with a particle of mass m delivers a momentum of $2mv_x$ to the wall. Only particles within a distance of τv_x will be able to hit the wall in time τ , so “collidable zone” thus has a volume of $\tau v_x A$. The number of particles in this collidable zone is given by the number density N/V times the volume of this zone. Putting these together, we find that the momentum delivered by particles of velocity component v_x , to a wall patch of area A in time τ , is

$$\begin{aligned}
 &= (\text{collidable volume}) * (\text{density}) * (\text{mom. per collision}) \\
 &= (\tau v_x A) \left(\frac{N}{V}\right) (2mv_x) \\
 &= 2mv_x^2 \frac{N}{V} \tau A.
 \end{aligned}$$

Next, we must integrate over the v_x s, weighted by their distribution. The range of integration is 0 to $+\infty$, since only particles traveling towards the wall will collide with

it. As a result, we end up with a factor of one-half compared to the mean squared speed, meaning that

$$(\text{Momentum delivered to wall}) = m \langle v_x^2 \rangle \frac{N}{V} \tau A.$$

To find the pressure, we find the force (momentum per time), and divide by the area:

$$P = \frac{(\text{Momentum delivered to wall})}{\tau A} = m \langle v_x^2 \rangle \frac{N}{V}. \quad (31)$$

We wish to write the mean squared velocity in terms of the temperature $k_B T$. To do this, recall the “equipartition theorem”: each component of the velocity, with kinetic energy $m \langle v_i^2 \rangle / 2$, has a thermal energy of $k_B T / 2$. We can also see this explicitly by performing the integral in Eq. 25. In any case, we see that $m \langle v_i^2 \rangle = k_B T$. Putting this into Eq. 31 yields the ideal gas law

$$\boxed{P = Nk_B T/V}. \quad (32)$$

E. Density Fluctuations: Experiment

To motivate the study of fluctuations, I would like to bring attention to the last panel of Fig. 2. The histogram of speeds does not perfectly overlap the theoretical shape, but is only close to the theoretical shape. Looking at some of the animations in <http://stanford.edu/~jeffjar/files/physics113/>, it is evident that systems do not evolve to equilibrium, but rather, they evolve *close* to equilibrium. To fully describe the behavior of systems, then, we must how close they lie to their equilibrium position.

Rather than studying the fluctuations of the speed distribution, I will study fluctuations in local density. Although it is possible to quantify the “distance” from the equilibrium distribution (for instance, with a χ^2 statistic), I will discuss fluctuations in density, because it is easier to think about the question “what fraction of the particles are on the left half of the box?”. As it turns out, density fluctuations behave similarly to other sorts of fluctuations, such as pressure fluctuations, or energy fluctuations in the canonical ensemble. Studying the “toy example” of fluctuations in density can help us think about the phenomena of fluctuations in general.

In particular, I will examine the behavior of the variable f , the fraction of particles on the left half of the box. The equilibrium value of f is 1/2, which we can derive from an entropic argument in the flavor of section IV B. The instantaneous value of f , however, may vary with time, because the particles are moving around. To observe this variation, I conduct experiments where I measure f as a function of time.

Shown in Figure 4 are experimental results, tracking f (the fraction of particles on the left half of the box) over time. The visual appearance of the plots makes it clear why we call this phenomenon “fluctuations”: as

time evolves, the value of f fluctuates around its equilibrium value of $1/2$. To quantify the magnitude of these fluctuations, we define the root-mean-square fluctuation $\Delta f \equiv \sqrt{\langle f - \langle f \rangle^2}$. (Again, I think that this definition is motivated by looking at the plot.)

Looking at the plots in Figure 4, we see that the magnitude of fluctuations Δf decreases as N increases from 10 to 50 to 100. To attempt to determine the dependence of Δf on N , I simulate gases with N varying from 10 to 150, and observe that Δf decreases, rather sharply at first, and then more slowly (bottom right panel).

To explain these observations about fluctuations, I will again discuss some standard statistical mechanical theory.

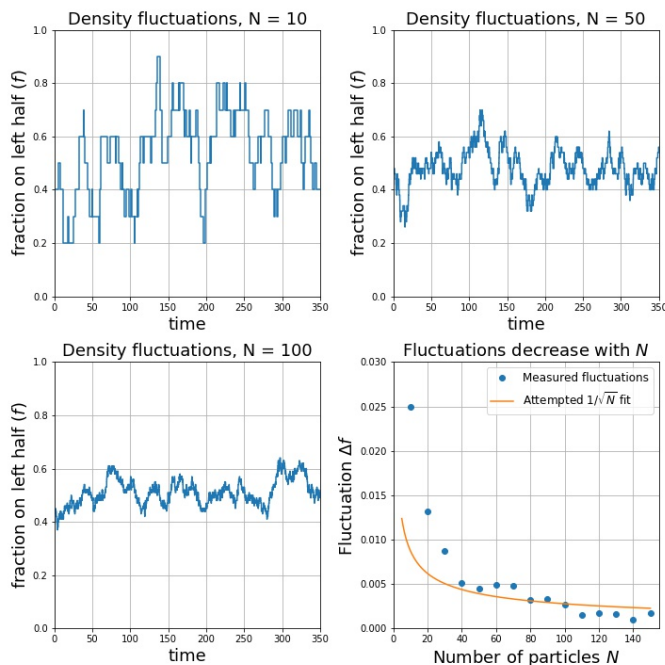


FIG. 4. The fraction of the particles on the left half of the box, f , fluctuates as a function of time. Shown above are the density fluctuations for $N = 10$, $N = 50$, and $N = 100$. Bottom right, the root mean squared deviation of the density, $\Delta f \equiv \sqrt{\langle f - \langle f \rangle^2}$, is plotted as a function of N . The fluctuations clearly decrease with N , but the attempted $1/\sqrt{N}$ fit is not very successful. All simulations are in 2D, with a volume of $V = 200^2$ and $k_B T = 20$. Fluctuations are averaged over 10000 events.

F. Density Fluctuations: Theory

There is a straightforward statistical argument that explains the trend between Δf and N , but the assumptions going into this argument (ie, independence, ergodicity and long-time averaging) are rather subtle. I will first state the argument, and then I will comment on its assumptions.

Suppose we model the positions of the N particles as following: (1) each particle has a uniform probability of being anywhere in the box, and (2) each particle's position is independent of the others'. Under these assumptions, each particle can be modeled as a random variable, with $1/2$ probability of being 1 (left half), and $1/2$ probability of being 0 (right half). (The expectation of this random variable is $1/2$, as is the standard deviation.) f is given by the average of N such random variables.

By the central limit theorem of statistics, when N is "sufficiently large," $\langle f \rangle$ is the expected value of the random variable, and Δf is its standard deviation divided by \sqrt{N} . In this case, the central limit theorem predicts $\Delta f = 1/(2\sqrt{N})$ – that fluctuations fall off as $1/\sqrt{N}$.

In the bottom left panel of Fig. 4, it is evident that a $1/\sqrt{N}$ curve does a poor job of fitting the experimental density fluctuations. A very plausible reason is that N is too small in these simulations. The central limit theorem only holds when N is sufficiently large that higher-order probability moments are insignificant. For instance, when N small and the predicted Δf is big, the f distribution definitely cannot be normal, since f must lie between 0 and 1.

Upon further inspection of the experimental data, however, we realize that the assumptions we make to invoke the central limit theorem are somewhat subtle. First of all, in the theoretical approach, we take a phase-space average of the particle position (that is, we consider the possible positions in the box) – but in the experimental observation, we take a *time* average of the particle positions. It is not obvious that time averages correspond to phase-space averages in any meaningful way. It takes careful thought to consider what this assumption really means. By explicitly tracing time evolution, as in this simulation, we can glean a bit of intuitive insight on this question of ergodicity.

Furthermore, the experimentally measured fluctuations are not *independent* samples from a phase space distribution – observations at time t are *correlated* with observations at time $t + \Delta t$. We can clearly see this on the plots in Figure 4: if we move a small distance along the x-axis, the fluctuation f tends to change less than if we move a large distance along the x-axis. In the parlance of non-equilibrium statistical mechanics, we have a nonzero correlation function $g(t) = \langle x(t)x(0) \rangle$. Our samples of f are clearly not independent.

There is a timescale τ associated with the correlation in f , however, and if we measure f over a long enough time $T \gg \tau$, we can ensure that we are sampling a true time average. (This timescale τ is related to how fast the correlation function decays to zero.) For instance, in the bottom left panel of Fig. 4, we see that the behavior of fluctuations do not fully reveal themselves after 50 units of time. Even with the 350 units of time shown in the

figure, it does not appear that we have fully thoroughly sampled the behavior of fluctuations. The fact that we sample f for too short of a time compared to its auto-correlation time τ , could explain why our Δf -to- N plot on the bottom right panel of Fig. 4 is not well described by a $1/\sqrt{N}$ curve.

To summarize the above discussion: the classic statistical argument to arrive at $1/\sqrt{N}$ fluctuations involves distributions over phase space, whereas the actual experimental observation involves distributions over *time*. Since measurements occurring closely in time are correlated with another, we must observe the system for much longer than the correlation time τ to correctly sample the time distribution. Finally, even if we obtain the true time distribution, we must make the ergodic assumption – that the laws of time evolution properly sample the phase space distribution – in order to apply the central limit theorem argument.

These arguments may seem rather theoretical and tangential, but I find them fascinating and rather crucial. Thinking carefully about these fluctuation-to-time plots gives us a glimpse into the marvelous world of non-equilibrium statistical mechanics.

V. EXTENSIONS

My event-based molecular dynamics engine can be easily extended to simulate related phenomena. Because my implementation has a modular architecture, users can easily add and remove features from the simulation. Here I briefly discuss one simple extension in the realm of non-equilibrium statistical mechanics.

A. Brownian motion

By placing a larger, more massive particle in the midst of the smaller gas particles, I can observe Brownian motion. Shown in Figure 5 is an example trajectory of a larger particle diffusing around inside the fluid. Notice that the diffusing particle appears to undergo a random

motion, even though the motion of every individual particle in the simulation is deterministic. With this visualization, we can explicitly see how the seemingly random behavior of diffusion emerges from the statistical effect of many moving particles.

The behavior of the diffusing particle can be modeled as a random walk, where the mean squared deviation from the starting position increases linearly with time. In particular, the trajectory $\vec{x}(t)$ can be modeled with

$$\langle |\vec{x}(t) - \vec{x}(0)|^2 \rangle = 2dDt, \quad (33)$$

where d is the dimensionality of the system and D is the so-called diffusion coefficient, with units of length squared per time [3]. I can verify this simple random-walk model by tracing the trajectory of the Brownian particle under different initial conditions. Shown in Figure 5 is the trajectory of the diffusing particle, with 10 different initial conditions. The squared deviation of the particle roughly increases linearly with time, suggesting that Eq. 33 is somewhat reasonable. With more simulations, I can explicitly measure the diffusion coefficient of the Brownian particle by fitting the average squared deviation to a straight line.

VI. CONCLUSION

In summary, I have explicitly demonstrated the core principles of statistical mechanics by simulating an ideal gas. I first demonstrate that the distribution of velocities quickly approach a Maxwell-Boltzmann distribution, regardless of their initial conditions. By seeing this system evolve to equilibrium, we see explicitly how systems evolve from low-entropy to high-entropy states. Next, I demonstrate the ideal gas law $PV = Nk_B T$ by measuring the pressure as a function of V , N , and $k_B T$, and find that my simulation agrees with the theoretical gas law. Finally, I show that fluctuations around equilibrium decrease with N , which explains why they are negligible in the “thermodynamic” large- N limit.

-
- [1] Severin Strobl, Marcus N Bannerman, and Thorsten Pöschel. Stable algorithm for event detection in event-driven particle dynamics: logical states. *Computational Particle Mechanics*, 3(3):383–388, 2016.
 - [2] S.K. Ma. *Statistical Mechanics*. World Scientific, 1985. ISBN 9789971966072. URL <https://books.google.com/books?id=3XyAOSYK4XIC>.
 - [3] G. Mazenko. *Nonequilibrium Statistical Mechanics*. Physics textbook. Wiley, 2006. ISBN 9783527406487. URL <https://books.google.com/books?id=koMHQPoyzBwC>.

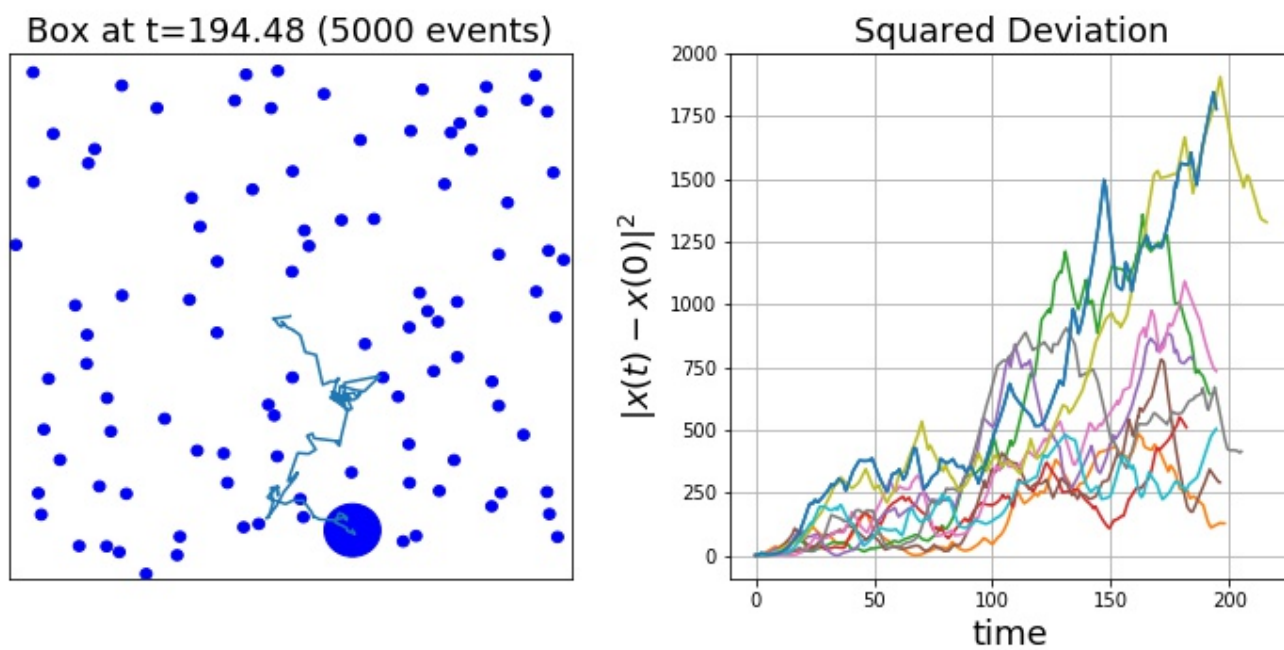


FIG. 5. The simulation engine can describe the Brownian motion of massive particles as they are buffeted around by the lighter gas molecules. Left, an example trace of the motion of a diffusing Brownian particle. Right, the squared deviation $|\vec{x}(t) - \vec{0}(0)|^2$ of the Brownian particle as a function of time, for ten different random initial seeds. Notice the roughly linear average increase. (The diffusing particle has 10x the mass and 5x the radius of the gas particles.)

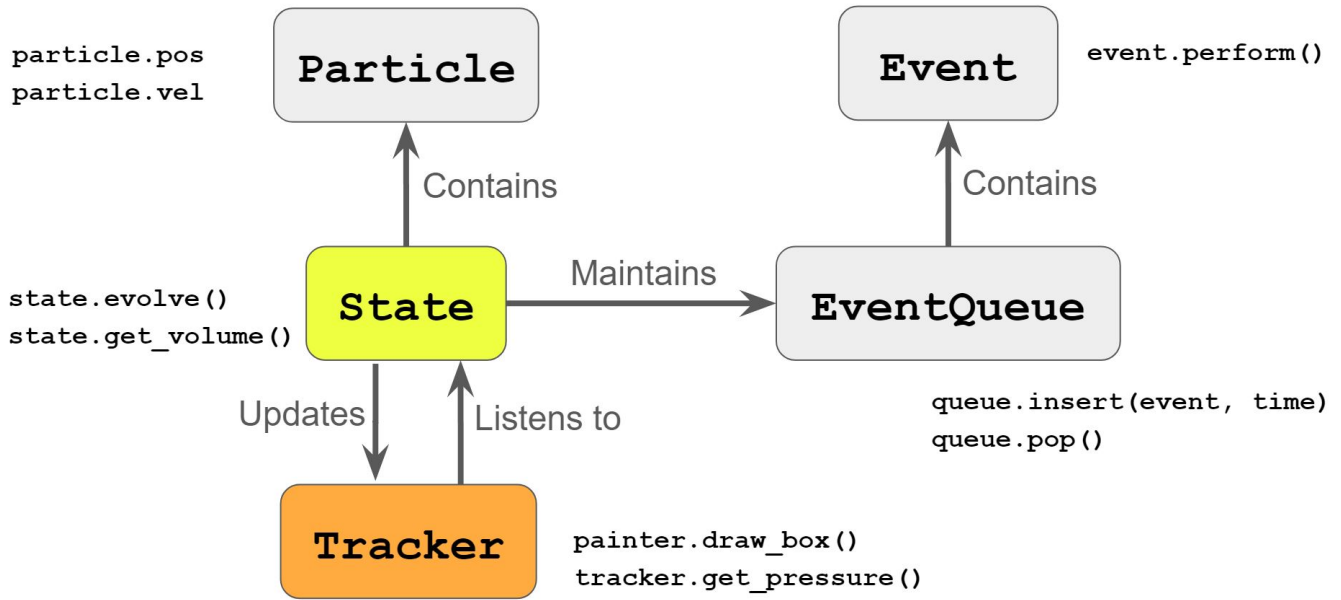


FIG. 6. Shown above is the software architecture of the code. The State object maintains a priority queue of Events, sorted by time. As the State evolves, the various Trackers monitor variables such as the pressure and total energy. Trackers also perform GUI tasks, such as drawing the state of the box, or displaying the distribution of speeds.