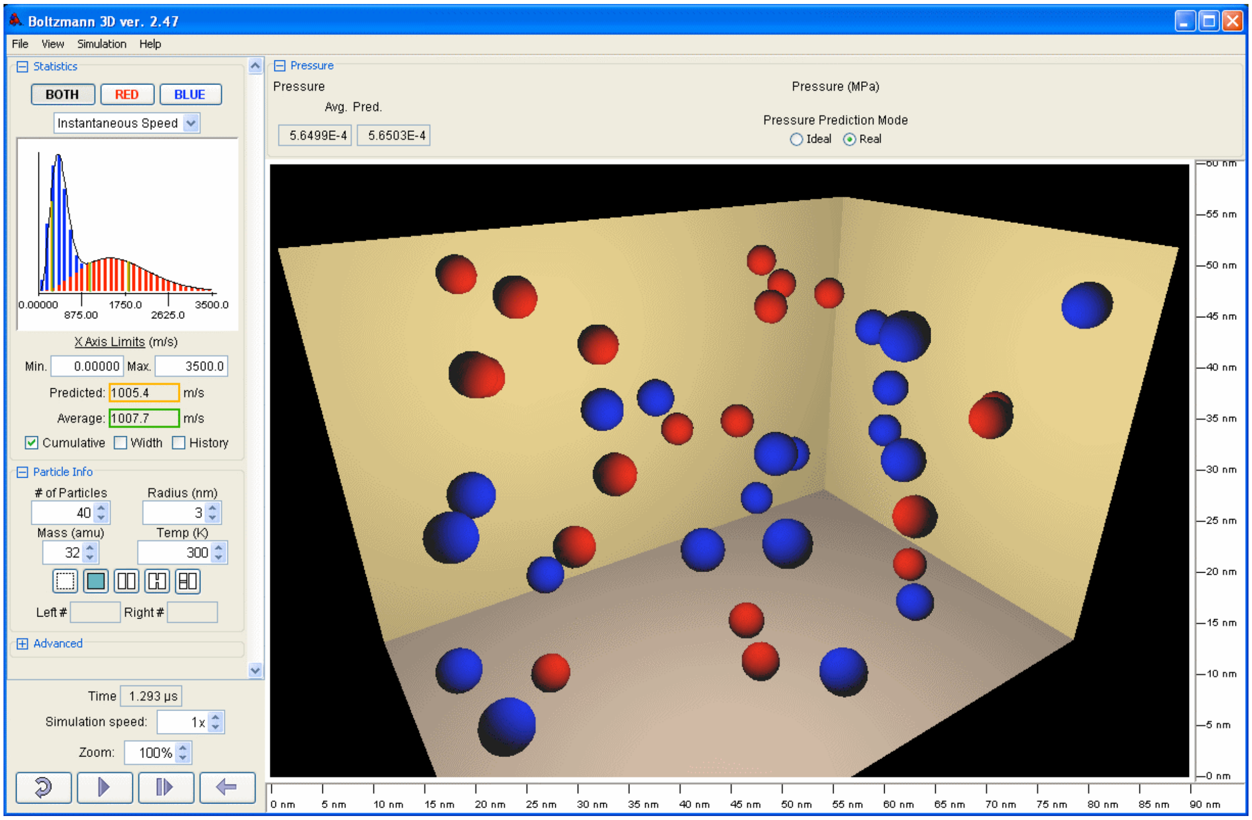
Boltzmann 3D Manual



Instantaneous Speed

Min. 0.00000 Max. 3500.0

Predicted: 1005.4 m/s

Average: 1007.7 m/s

☑ Cumulative ☐ Width ☐ History

# of Particles 40

Radius (nm) 3

Mass (amu) 32

Temp (K) 300

Left # Right #

Time 1.293 µs

Simulation speed: 1x

Zoom: 100%

Pressure Avg. 5.6499E-4 Pred. 5.6503E-4

Pressure Prediction Mode ○ Ideal ● Real

Please report problems or suggestions to:
Dr. Randall B. Shirts
Department of Chemistry
Brigham Young University
Provo, Utah 84602
801-422-4290
randy_shirts@byu.edu

Boltzmann 3D Manual Table of Contents

Boltzmann 3D

A Kinetic Molecular Motion Demonstrator

by
Randall B. Shirts, Scott R. Burt, Benjamin J. Lemmon, Jared D. Duke, and Dustin A. Carr
Department of Chemistry and Biochemistry
Brigham Young University
Provo, UT 84602

What does the motion of atoms and molecules look like? **Boltzmann 3D** is designed to demonstrate principles of molecular kinetic theory, statistical mechanics, and other fundamentals of chemistry and physics in a visual and understandable way. **Boltzmann 3D** simulates the motion of particles in a small box. You control the number, size, mass, and temperature of the particles and watch them move around in the box just like atoms move in a gas. You can perform hundreds of experiments to test interesting hypotheses or answer difficult questions:

- Do the particles obey the celebrated ideal gas law?
- Are the laws of molecular kinetic theory (upon which some of the most important principles of chemistry and physics are based) really true?
- How do particles chaotically and randomly collide with each other and with the walls of their container?
- How do simple laws result in complex motion?
- How does the same complex motion result in simple laws describing bulk properties such as the pressure exerted by a gas, the probability of evaporation or condensation, and many other important characteristics of real materials?
- How can simple colliding balls help you understand kinetics and equilibrium?

The motion of molecules determines almost everything around us in the physical world. You can learn and appreciate this using **Boltzmann 3D**. Just follow the bouncing balls!

Using Boltzmann 3D

Boltzmann 3D was written to help students understand the concepts of molecular kinetic theory. Many of the properties of matter (pressure, evaporation, melting, vapor pressure, phase behavior, reaction rates, equilibrium, etc.) depend on the distribution of molecular speeds. Experience shows that when you just *tell* students about these concepts, they don't really understand. If you help them visualize relative velocities, they understand the concept at a much deeper level.

Boltzmann 3D makes an excellent classroom demonstration tool when projected on a large screen. A good demonstration would be to start with 50 to 100 randomly distributed particles and watch the velocity distribution develop from a single initial velocity. Using red and blue particles with different masses, you can also show how mass affects velocity. Effusion can be demonstrated using a divided arena. Each demonstration can be saved in a separate *.set file so you can quickly load each file without disrupting the flow of your class. Some example *.set files are supplied with the program.

An even more powerful use of **Boltzmann 3D** is to make the program available to students to play with on their own. They can change the temperature and the mass of particles themselves and observe the resulting changes. Give them a set of questions and a worksheet to guide them. Then, let them explore.

Hardware Requirements

Boltzmann 3D was designed to be platform and system independent. All it requires is:

- Windows/Linux: 700MHz Pentium 3 processor or compatible, Mac: 700 MHz G3 processor. Larger and more complex simulations require faster processors. The program may work on some older, slower computers, but you will have to use fewer particles (try 30 to 40), eliminate shading, and use 2D rather than 3D simulations.
- An OpenGL compatible graphics card. Faster graphics will make the motion appear more realistic. An older graphics card may not be able to use the most recent graphics libraries and may offload some computation to the CPU. This will interrupt motion and make it appear jerky. Before you download **Boltzmann 3D**, you should update your graphics drivers by going to the manufacturer's website (see below under graphics drivers).
- Java 2 SE Runtime Environment 1.4.2 or later. The latest version can be downloaded from http://www.java.com (or **software update** for Mac users).
- OpenGL 1.5 or later graphics drivers. The best way to update your graphics drivers is to go to the website of your video card or computer manufacturer and download any updated drivers. For a Windows computer, find the video card make and model by right-clicking on the desktop, selecting **properties** and then the **settings** tab, and then looking under the line **display**. For Macs, go to **software update**.

Obtaining Boltzmann 3D

Download the Java code, documentation, sample demonstration files and this manual by going to http://people.chem.byu.edu/rbshirts/research/boltzmann_3d and clicking on the link corresponding to your operating system: Windows, Mac, or Linux. Graphics drivers for Linux are somewhat non-standard and may require customization.

Starting Boltzmann 3D

After installing the software, start **Boltzmann 3D** by choosing one of the following:

- Windows: double click Boltzmann.jar or the **Boltzmann 3D** icon if you have elected to have it on your Windows desktop
- Mac: double click Boltzmann 3D.app
- Linux: run /usr/bin/Boltzmann3D

Boltzmann 3D will begin to load. After a few seconds, the **Boltzmann 3D** window will appear, and the program will start a simulation using the default settings.

Screen Resolution

Boltzmann 3D can run with just about any screen resolution. Recommended resolution is 1024×768 or higher. At low resolutions like 800×600 or 640×480, you will need to resize the arena so it fits on your screen and use smaller particles. At high screen resolution, you may wish to make the arena larger and use larger particles. You can resize the arena in the **options** under the **simulation** menu or by dragging the corner. The following table gives approximate horizontal and vertical sizes for full-screen operation in several common screen resolutions. If you have a visible task bar (or dock for Mac), subtract 0.1 nm for every pixel it occupies. If you prefer, you can scale the sizes of everything in the arena by adjusting the **Zoom** option.

horizontal pixels	vertical pixels	arena width (nm)	arena height (nm)
640	480	32.7	29.2
800	600	48.7	41.2
1024	768	71.1	58.0
1280	1024	96.7	83.6
1600	1200	128.7	101.2

I. Boltzmann 3D Main Window

Boltzmann 3D represents the motion of particles graphically by moving circular spheres (simulated atomic-size particles) around on the computer screen. What you see in **Boltzmann 3D** is only a simplified model of real atomic behavior. In fact, real atoms are *much* more complicated than circular spheres. Scientists call this kind of model a *simulation*. The motion of particles in **Boltzmann 3D** is both a mathematical simulation taking place inside the computer using Newton's laws of motion and a graphical simulation displayed on the computer screen. At the lower left of the window is a clock that tells how much time has passed since the start of the experiment. Because the motion of atoms is incredibly fast, this clock is adjusted to run correspondingly slow. In one second of time on your wall clock or wrist watch, you will see the complicated motions that occur in just a few picoseconds (1 ps = 10^{-12} s) of simulation time.

The **Boltzmann 3D** main window is divided into six key areas: the **arena**, the **pressure display** area, the **statistics** area where distributions are displayed, the **particle information** area, the **advanced** area, and the **program control** area. The pressure display area, statistics area, particle information area, and advanced area can be hidden by clicking on the **–** sign by the title or expanded by clicking on the + sign by the title.

A. The Arena

Most of the **Boltzmann 3D** window is occupied by the arena in which particles move and collide with each other or with the arena walls. The size of the arena is adjustable. Resize it by grabbing with the mouse at the lower right corner or by setting the dimensions in **options** under the **simulation** menu. The size is labeled in nm (nanometers, 1 nm = 10^{-9} m). Although a region at the nanometer scale is extremely small, particles moving in the arena still exhibit many of the same properties and principles that real atoms exhibit. Particles move in the arena at a constant velocity in a constant direction until they collide with either another particle or one of the walls of the arena. You can change the number of particles, or their radius, mass, and temperature (or average kinetic energy) using controls in the **particle information** area described below.

In **Boltzmann 3D**, particles do not have any attractive forces between them; they elastically rebound when they collide. Each particle has a definite radius, and a collision occurs whenever the centers of two particles are at a distance given by the sum of the particle radii. A collision with the wall occurs when the center of a particle is at a distance from any wall equal to the particle radius. The velocities and directions of both particles after a collision are calculated such that the total momentum and kinetic energy are the same as before the collision. Thus, the only principles assumed are conservation of momentum and energy. All of what you see follows directly from these two universal principles of physics.

A simulation can be performed in one dimension (1D), two dimensions (2D) or three dimensions (3D). Because the computer screen is two dimensional, the most useful choice is to do simulations in two dimensions (horizontal and vertical). The dimension can be changed from the **simulation** menu under **dimension**. One-dimensional simulations confine the spheres to a horizontal line across the arena. Three-dimensional simulations are more difficult to portray on a 2D computer screen; as a particle moves away from the viewer, its size shrinks and the shading on the particle shifts. In addition, you can tilt your perspective of the arena by clicking on the horizontal or the vertical ruler to activate the window area and then using the directional keys right of the keyboard. Some of the consequences of the laws of physics are slightly different depending on whether a simulation is in 1D, 2D, or 3D. You can still understand many of the important principles of three-dimensional particles by observing two-dimensional ones, but the subtle differences can be instructive.

The arena can be divided into two equal regions by clicking on the corresponding icon in the **particle information** area. Use this mode to show the differences between two sets of particles (red and blue), each with its distinct mass, size, or temperature. For 2D or 3D simulations, a small hole can be opened between the two regions to observe effusion or mixing. Open the hole by clicking on the icon to the right of the **Divided Box with Walls** icon. You can also implement a **piston** using adiabatic or isothermal conditions to study compressions or expansions.

Choose **periodic boundary conditions** in a **Boltzmann 3D** simulation by clicking the box with a dotted line around it in the **particle information** area. Periodic boundary conditions mean that there are no walls around the arena. Think of identical copies of the simulation adjacent to each other on all sides. When a particle leaves through the right side of the arena, an identical particle from the copy immediately on the left enters from the left. Likewise, a particle leaving on the left is replaced by one entering from the right. Particles can also leave or enter through the arena's top or bottom. In 3D, particles can also leave and enter from the front or back; although in 1D, they can only leave and enter from the left and right. Using periodic boundary conditions eliminates the effect of walls on the system while keeping the number of particles constant.

Even so, there are still subtle differences between a small system of a dozen or even a thousand particles and one with an infinite number of particles. You cannot choose a divided arena, a small opening between divided regions, or a piston while using periodic boundary conditions.

Some aspects of kinetics and equilibrium can be studied by selecting **reaction mode** in the **simulation** menu.

B. The Pressure Area

Just above the arena are two display boxes and two radio buttons. The two display boxes on the left show the **average pressure** and **predicted pressure**. The radio buttons allow you to select how the predicted pressures are calculated, i.e. using either real gas or ideal gas predictions. When the arena is divided, the two display boxes on the left show the **left wall average pressure** and **left wall predicted pressure,** and two additional display boxes on the right show the **right wall average pressure** and **right wall predicted pressure**.

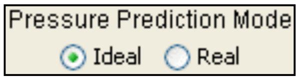
For a small number of particles with small radius, the ideal gas equation ($PV = nRT$ in 3D, $PA = nRT$ in 2D, and $PL = nRT$ in 1D) relates the pressure to other useful quantities. Pressure, P, is the force exerted per unit wall area for a 3D volume, V, or force per unit wall length for a 2D area, A, or average force on the wall for a 1D length, L. In all of these equations, n is the number of moles of gas, T is the absolute temperature, and R is the universal gas constant. One of the assumptions of the ideal gas law is that the volume (or area in 2D, length in 1D) occupied by the particles is negligible. If you choose **Ideal** predictions, all predictions will be made neglecting the volume of the particles compared to the volume of the arena. Note that units for pressure in 2D are expressed in force per unit length (Newtons per meter). Values are displayed in micro-Newtons per meter (abbreviated μN/m). In 1D, force on the wall is displayed in picoNewtons (pN), and in megapascals (MPa) in 3D.

If you select the **Real** predictions button, the pressure will be calculated using a more accurate formula derived from extensive theoretical calculations and computer simulations as discussed in Section V.

Each time a particle collides with one of the walls, an instantaneous force is exerted on the wall and, conversely, by the wall on the particle. The momentum transferred over a period of time is averaged to determine the pressure exerted by these collisions. Time averaging is required because the force is infinite at the instant of a wall collision but is zero at all other times.

Each collision with the left or right wall transfers momentum $2m|v_x|$ to the wall. To give a useful average over a reasonable period of time, these quantities are accumulated for the pressure averaging time, T_{avg}, and the sum is divided by the averaging time and displayed in the boxes labeled **Avg**. The default value for T_{avg} is 60 ps. Predicted pressures are to the right of the calculated time averages. The predicted values may be calculated using either the ideal gas equation or a real gas law (click on the **Ideal** or **Real** radio button above the arena). Displayed pressure or pressure history during the first 60 ps after a new start will not have normal behavior. The default value of T_{avg} (60 ps) may be changed to another value in **options** under the **simulation** menu. A larger T_{avg} will give smaller fluctuations in displayed pressures, but pressures will not respond as quickly to changes in the system.

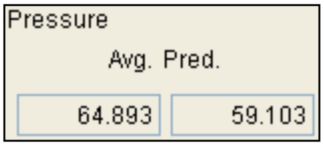
Ideal/Real Prediction Indicator


Pressure Prediction Mode

● Ideal ○ Real

Click on **Ideal** to use the ideal gas law predictions or click on **Real** to use more accurate real gas law predictions based on computer experiments which take the non-zero size of the particles into account.

Wall Pressure Gauges (Avg.)


Pressure

Avg.	Pred.
64.893	59.103

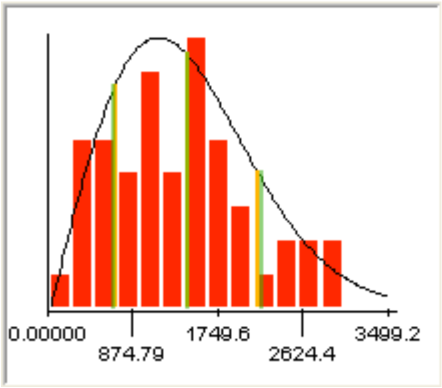
This box displays the pressure (in picoNewtons for 1D, microNewtons per meter for 2D, megaPascals in 3D) due to collisions of particles with the left and right walls. The value displayed in this box is averaged over 60 ps of motion (or other averaging time set in **options** under the **simulation** menu). When using periodic boundary conditions, the box is labeled **virial pressure**, and displays the pressure calculated using the virial theorem. When the arena is divided, boxes are displayed for both **left and right wall pressure**.

Wall Pressure Prediction (Pred.)

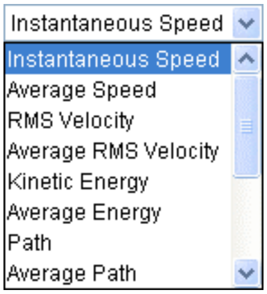
This box displays the predicted pressure on the left and right walls (in picoNewtons in 1D, microNewtons per meter in 2D, or megaPascals in 3D). This prediction can be based on the ideal gas law or a real gas law correcting for the finite size of particles. You can change between these using the **Ideal/Real** buttons. When the arena is divided, two predictions are displayed that depend on the number of particles on the left side and right side of the arena (left number or right number boxes are displayed in the **particle information** area).

If you have two types of particles moving in the arena (red or blue), each type produces its *own* contribution to the pressure when it collides with the walls. You can see these contributions, called, *partial pressures*, by clicking on **red** or **blue** in the statistics area. The total pressure is displayed when **both** is clicked.

C. The Distribution Graph Display (or Statistics) Area



At the top left of the window is a dynamic graph which can display any of many different plots. Use the **plot selection pull down menu** above the graph to change the current plot. These statistical graphs chart the behavior of the particles in the arena. The abscissa (*x*-coordinate) range is divided into 15 regions with colored bars indicating the number of particles in each subrange of the displayed quantity. This is called a histogram—a type of bar graph. The green vertical lines displayed on the graph represent the calculated average of the quantity displayed and the average plus or minus one width (root-mean-square deviation). The orange vertical lines show the predicted values of the average quantity and that average plus or minus one width. The heights of these lines are approximately what the histogram heights would be expected to be. These values are also shown in text boxes below the plots. Plots also show a black curve indicating the predicted distribution.



Plot Selection Pull Down Menu

Select the currently displayed plot from the **Plot Selection Pull Down Menu**. The available plots depend on the current dimension of simulation and the boundary mode. The possible plots and an explanation of each are listed below. You can use the + and − keys to move up and down in this menu list (keyboard or number pad).

Instantaneous Speed Distribution

When you click on the **Restart** button, all particles start with random positions and directions but with identical velocities, so you begin with only one histogram bar. After a few collisions, the distribution widens out to the Maxwell-Boltzmann distribution, which is shown as a black line. The quantity plotted is $(v_x^2 + v_y^2 + v_z^2)^{1/2}$ in 3D, $(v_x^2 + v_y^2)^{1/2}$ in 2D, and $|v_x|$ in 1D. The average and average plus and minus one root-mean-square deviation are shown by the green lines.

Average Speed Distribution

This histogram shows the average speed of each particle (total distance traveled divided by total time) since the last **Restart** command. The speed of a particle changes each time it has a collision with another particle. At any one time, particles will have a wide distribution of speeds, but over a long time, the average speed of each particle can be seen to approach the same average. That is, the width of the distribution of average speeds decreases with time after the first few collisions. This is an example of a famous principle in statistical mechanics called the *ergodic hypothesis*.

Root-Mean-Square (RMS) Velocity Distribution

This histogram is identical to the instantaneous speed plot except that the quantity averaged for the vertical lines and displayed values is $v_x^2 + v_y^2 + v_z^2$ in 3D, $v_x^2 + v_y^2$ in 2D, and v_x^2 in 1D, after which the square root is taken. The average squared velocity of many particles with the same mass is proportional to the temperature, so the average value of this quantity should be constant if energy is conserved. (If you have particles with different masses, however, the RMS velocity may not be exactly constant.)

Square Root of Time-Averaged Squared Velocity Distribution (Average RMS Velocity)

At any one instant, not all particles have the same squared velocity (there is a distribution of these values), but as time increases, the width of the distribution of time-averaged squared velocity gets smaller. This is another example of the *ergodic hypothesis*. This distribution shows the square root of that quantity. The average should approach the RMS velocity.

Kinetic Energy Distribution

Just as velocity changes in time, the kinetic energy, $mv^2/2$, of a particle changes. This histogram shows how many particles have kinetic energy within different ranges at any instant. The predicted curve is called the Boltzmann distribution.

Average Kinetic Energy Distribution

At any one instant, particles have a distribution of kinetic energy values. This distribution contains the time-weighted average kinetic energy for each particle. The width of the distribution of average kinetic energies should decrease with time to the average kinetic energy, yet another example of the *ergodic hypothesis*.

Free Path Distribution

The free path is the distance traveled by particles between collisions with other particles. Collisions with walls are not considered, only collisions between two particles. The path lengths shown are paths for each particle between the next-to-most-recent collision and most recent collision. Because collisions between particles occur randomly, some particles travel a short distance and some may travel a long distance. This distribution is predicted to be an exponential distribution, so the average (mean free path) and the width should be identical (see Section IV).

Average Path Distribution

This plot shows the average path between collisions for each particle separately averaged over time since the simulation started (in other words, the total distance traveled divided by the number of collisions). Only collisions between two particles are considered. As time increases, the width of the distribution should become narrower as predicted by the *ergodic hypothesis*.

Time per Collision Distribution (Lifetime)

This plot shows the time between collisions with another particle. The values are times for each particle between the next-to-most-recent collision and the most recent collision. Because collisions between particles occur randomly, some collide quickly and some may go a long time between collisions. The distribution is predicted to be an exponential distribution, so the average (also called the collision lifetime or just the lifetime) and the width should be identical (see Section IV).

Collision Rate Distribution

The collision rate is the number of collisions per unit time that each particle has experienced with other particles since the simulation began. Each time a collision occurs, the collision is counted twice (once for each particle) and a record kept. For long times, the distribution of collision rates has a width that decreases, yet another example of the *ergodic hypothesis*.

Radial Distribution Function (RDF)

This distribution shows the probability, given the position of one particle, that another particle is at a distance *r* relative to what would be expected for a uniform fluid. Because of the effect of walls, this distribution is only of use when using **periodic boundary conditions**. Because only the nearest of several possible periodic image particles is counted, the maximum abscissa value should be limited to half the minimum arena dimension (default). This function has a constant value for distances greater than the particle diameter for a dilute gas. At higher densities, oscillations develop that show the structure of the fluid. The ratio of the value at the particle diameter compared to that at large distances indicates by what factor the real pressure deviates from the pressure of an ideal gas at the same density. When red and blue particles are both included in the simulation, separate RDF distributions are calculated for red-red distances (select **red** indicator), blue-blue distances (select **blue** indicator) and red-blue distances (select **both** indicator).

X, Y, and Z Velocity Distributions (X-, Y-, or Z-Component of Velocity)

The Maxwell-Boltzmann distribution predicts a bell-shaped curve (Gaussian or normal distribution) with an average velocity of zero for any component of velocity. The long-time average must, of course, be exactly zero because the arena is not moving, but the instantaneous average will fluctuate about zero. The width of the distribution depends on the temperature and particle mass. The *y*-component graph is only available in 2D and 3D simulations; the *z*-component graph is only available in 3D.

Number of Particles Graph

Only available in **reaction mode**, two lines show the number of red (reactant) and blue (product) particles as a function of time for the previous 150 frames of motion.

Wall Pressure Graph (either Left or Right with a divided arena)

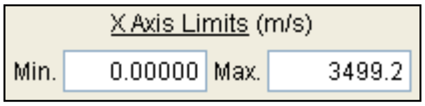
The pressure on the left wall and right wall are averaged over 60 ps of motion (unless the default is changed using **options** under the **simulation** menu), averaged together, and displayed. Separate left and right wall pressure graphs are available with a divided arena. If the averaging time is too short, the pressure fluctuations are inconveniently large; too long, and the pressure does not respond quickly to changes made in the simulation. These graphs only display in history mode because there is no distribution to display. Predicted pressures may be calculated using either the ideal gas equation or a real gas law (click on the **Ideal** or **Real** radio button above the arena). These graphs are only available when walls are present.

Virial Pressure Graph

When periodic boundary conditions are used, the pressure is calculated using a relation derived from the virial theorem. This pressure does not fluctuate as much as wall pressure because it uses all simulation data since the last **Restart**.

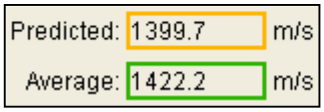
Changing the Maximum or Minimum Abscissa of a Plot

For each plot, the abscissa (x-axis) is divided into 15 equal regions (30 regions for cumulative distributions), and the number of particles with the chosen quantity falling within each region is shown as a histogram. The maximum and minimum values of the abscissa are both displayed in text boxes below the statistics plot. To change the maximum or minimum from its default value, highlight it and type a new value. If the entered maximum is less than the present minimum, the new maximum will be set to be 110% of the minimum. If the entered minimum is greater than the present maximum, the new minimum will be set to be 90% of the maximum. Each distribution has its own maximum and minimum, and these are saved in a set file.


X Axis Limits (m/s)
Min. 0.00000 Max. 3499.2

Predicted Value

This text box displays the theoretically predicted average quantity of the currently displayed graph. (See Section IV for the theory concerning each of these predictions.) You may also display the predicted width (root-mean-square deviation) of the distribution by clicking the **Width** box. When predicting pressures, if you have chosen **Ideal** predictions, the predictions will be based on the ideal gas law, but if you have chosen **Real** predictions, the predictions will be corrected for

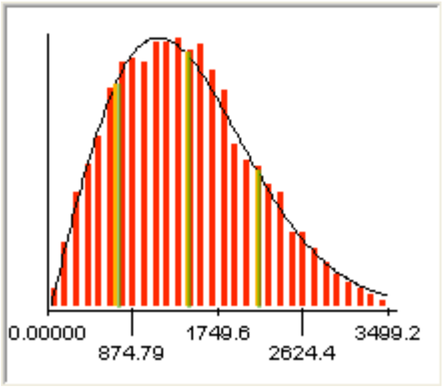

Predicted: 1399.7 m/s
Average: 1422.2 m/s

the finite size of the particles. Change between the two by clicking on the **Ideal** or **Real** radio buttons in the pressure area. Predicted values are those for an infinite system unless the **Include finite system corrections** option is selected from the **Simulation** menu.

Computed Average Value

This text box displays the average of the quantity that you select, based on the simulation in the arena. If you have the cumulative box checked, the average will be for the cumulative distribution (see below) displayed. Otherwise, the box displays the average of the currently displayed instantaneous graph. You may also display the computed width (root-mean-square deviation) of any distribution by clicking the **Width** box.

Displaying Cumulative Distributions (Cumulative Box)

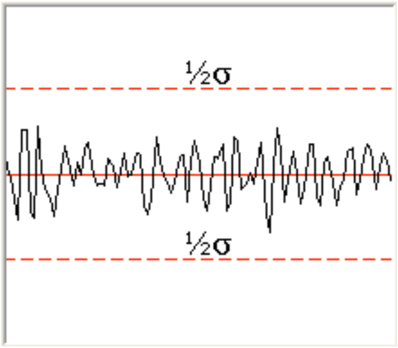


The histogram normally displays information from a single instant of time. If you wish to see the cumulative distribution containing all recorded events since the simulation started, check the cumulative box. The cumulative distribution has thirty bars rather than fifteen, and, while it shows the same type of information as the instantaneous distribution, it changes more slowly and exhibits smaller fluctuations. This feature allows you to see if the distribution of lifetimes or velocities really matches the predicted distribution. The cumulative distribution is updated each time a collision occurs and shows how often any particle possessed the chosen quantity since the histogram was zeroed. To zero the cumulative histogram, click on **Restart** to zero all cumulative histograms, or change the maximum or minimum of a histogram to re-zero just that one histogram. All the cumulative distributions are zeroed automatically at a time four times (or some other user chosen multiple; see **options** in the **simulation** menu) the predicted average collision time after a **Restart**. We do this to eliminate any values that might depend on the way initial values were chosen. You should not pay any attention to the cumulative distributions for average speed, average RMS velocity, average kinetic energy, average path, or collision rate because these distributions change with time. These are left in the program because it is faster for the computer to compute them than to be continually checking which cumulative distributions should be left out. Velocity and energy cumulative distributions are weighted by time duration with each value. Path, time, collision rate and radial distribution function are weighted only by the number of occurrences of that value. When the cumulative box is checked, the displayed **Average** and **Predicted** values are for the cumulative distribution. When the cumulative box is not checked, the displayed values are for the instantaneous distribution.

Displaying Distribution Widths (Width Box)

Click the mouse on the **Width** button to display the calculated and predicted distribution widths — the root-mean-square (RMS) deviation of the distribution ($\text{width}(q) = \text{Sqrt}[<q^2> - <q>^2]$, where q is any quantity, and $<q>$ is an average over the N particles). This quantity is also called the *dispersion* and is approximately the *standard deviation* if the number of particles is large. The predicted values for some quantities (pressure, mean free path, collision time and collision rate) depend on whether you assume the particles to be an ideal or real gas. Uncheck the **Width** box to return to displaying the average quantities.

Displaying Histories (History Box)

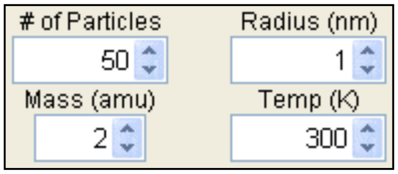


Check the **History** box to see the history of values for the most recent 150 frames for any of the quantities plotted. The predicted value is shown as a red line and one half of the predicted rms deviation (or width) is added or subtracted to get the dotted red lines and the total vertical scale is four times the predicted width. For pressures, the width is taken to be the pressure itself so the scale goes from zero to twice the predicted pressure. Viewing a history will give you an idea of how much the quantity is changing or fluctuating. There is no distribution defined for left pressure, right pressure, or virial pressure, so if you select one of these distributions, the plot will automatically display history. Return to a distribution by unchecking the **History** box.

Red, Blue, and Both Indicators

Boltzmann 3D can keep track of two different colors of particles. There are three choices for displaying distributions and averages of the simulated particles. You can display the attributes of just the **Red** set, just the **Blue** set, or **Both** sets. Predicted average quantities for the red and blue sets of particles and the averages calculated from the simulation can be viewed by clicking the corresponding button at the top of the **Statistics Area**. Number-weighted average quantities for both sets are viewed by clicking the button marked **Both**. All particles are red at the beginning of a simulation. You cannot change the number of particles when the Red set indicator is on because all particles created after the start of the simulation are blue. Blue particles can be added by pressing the number + button, by clicking the mouse at a position where you would like to add the particle, or by typing a new number in the particle number indicator. If there are both red and blue particles in the arena, they can be the same, or they can differ in mass, radius, and/or initial temperature. You can have any number of red and blue particles up to a total of 999. The mass and radius of a particle remain constant, but the temperature doesn't because particles exchange energy when they collide. For example, if the blue particles have a higher temperature, in other words, a higher average kinetic energy, then after many collisions (several collision lifetimes) the blue particles should cool down and the red ones should heat up. You can check this by converting from kinetic energy to temperature using $KE = Mv^2/2 = dRT/2$ where d is the dimension (1, 2, or 3).

D. The Particle Information Area



# of Particles 50
Radius (nm) 1
Mass (amu) 2
Temp (K) 300

You control the number, mass, radius, and temperature of particles in the arena using the **particle information** area buttons. You can also divide the arena to see effusion and other phenomena.

Number of Particles

Change the number of particles by clicking on the increase or decrease buttons to the right of the **Number of Particles** box or by highlighting the number in the box and typing a new number. The total number of particles may be in the inclusive range 0 to 999. If the particles do not conveniently fit in random positions, the number will be automatically adjusted downward. Any particles defined when **Boltzmann 3D** starts or when you click the **Restart** button are red. Additional particles added by increasing the number in this text box are blue. You may also add one blue particle at a time by clicking the mouse at any spot within the arena. Decreasing the number in the text box removes the specified number of particles from the simulation. The most recently created particles will be removed first. Note that removing particles will change the temperature of remaining particles depending on the instantaneous kinetic energy of the particles removed.

Mass of Particles

Change the mass of particles to be added afterward by clicking on the increase or decrease buttons to the right of the **Mass** box or by highlighting the value in the box and typing in a new value in the inclusive range 0 to 999. To change the mass of all particles, change the mass to the desired amu value and click on **Restart**. To add blue particles of a second mass, change the displayed mass value and either click the mouse within the arena or click the **Number of Particles** increase button. Any new blue particles will have the new mass and the old red particles will remain with their original mass. You may add blue particles of several different masses, but the correct averages and distributions will be predicted only if all blue particles have the same mass.

Radius of Particles

Change the radius of particles by clicking on the increase or decrease buttons to the right of the **Radius** box or by highlighting the value in the box and typing in a new value. The radius must be at least 0.2 nm and no greater than 20 nm. To change the radius of all particles, change the radius to the desired value and click on **Restart**. To add particles of a second radius, change the radius to the desired value and either click the mouse within the arena or click the **Number of Particles** increase button. Any new blue particles will have the new radius. You may add blue particles of several different radii, but the correct averages and distributions will be predicted correctly only if all blue particles have the same radius.

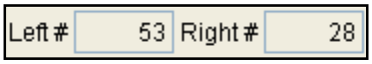
Temperature of Particles

Change the temperature of particles (in degrees Kelvin) by clicking on the increase or decrease buttons to the right of the **Temp** box or by highlighting the value in the box and typing in a new value in the inclusive range 0 to 9999. To change the temperature of all particles, change the temperature indicator to the desired value and click on **Restart**. Any blue particles you add will have the same temperature as the displayed value. Add blue particles with a different temperature by changing the temperature indicator and either clicking the mouse within the arena or clicking the **Number of Particles** increase button. If the temperature of blue particles is different than that of the other particles, it will be forgotten as energy is shared among the particles during collisions. In the process, the average energy and temperature of all the particles will change. When you add blue particles of a different temperature, **Boltzmann 3D** calculates the total energy, calculates a new temperature, and alters the number in the temperature indicator to the new, actual temperature. However, when using **piston mode** or **reaction mode**, the particle information area temperature indicator retains the value it has at the beginning of a simulation. In **piston** mode, the piston area displays its own instantaneous temperature indicator.

Boundary Mode Icons (Dividing the Arena)

The arena has five **boundary modes** with corresponding icons in the **particle information** area. They are (from left to right): **Periodic Boundaries, Box with Walls, Divided Box with Walls**, **Divided Box with a Hole** and **Piston** mode. The default mode is **Box with Walls**. When the **Periodic Boundaries** mode is selected, any particles that leave the edge of the arena come back in on the opposite end of the arena. This is particularly useful when simulating an infinite system because there are no walls. In all other modes, the particles will bounce off the walls. In **Divided Box with Walls** mode, a wall is placed in the middle of the arena dividing it into two equal halves. Switching to divided mode will automatically restart the simulation with all the particles on the left side of the divider. To study effusion, switch to the **Divided Box with a Hole** mode. The diameter of the hole can be set in the **options** dialog under the **simulation** menu. Certain predicted values may not be correctly predicted when the arena is divided because the predictions are calculated as if the arena is not divided. Use **Piston** mode to do either adiabatic or isothermal compressions or expansions to see changes that take place.

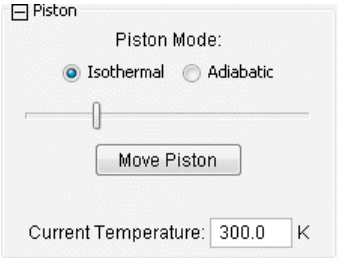
Left and Right Number Text Boxes



Left # 53 Right # 28

The **Left** # box displays how many particles are in the left side of the arena. The **Right** # box displays how many particles are in the right side of the arena. These boxes only function when the arena is divided.

Piston Controls



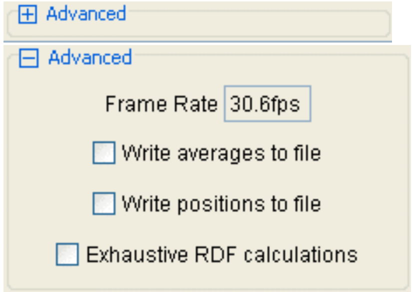
● Isothermal ○ Adiabatic

Current Temperature: 300.0 K

If you choose **Piston** mode, a new set of controls pops up below the boundary mode icons. Move the slider to the desired final volume and click the **Move Piston** button to start a piston compressing a gas in the arena. You can choose either **adiabatic** or **isothermal** conditions by checking the appropriate radio button. Adiabatic conditions means no heat is transferred in or out of the system. A moving piston imparts an increment of momentum to a colliding particle. In other words, work is done by the piston in a compression, and this raises the temperature of the gas as shown by the current temperature display (the original temperature in the particle control area is left unchanged for comparison). Isothermal conditions means that energy is subtracted from the particles or added to the particles in each wall collision to maintain a constant temperature as the piston moves. To expand the gas after a compression, reset the piston slider to a larger volume and click **Move Piston** again. The rate of piston motion can be adjusted in the **simulation** menu. During isothermal changes, the words “Thermostat active!”

will flash below the piston button as long as collisional kinetic energy is being adjusted to keep the temperature constant. The maximum percent change in kinetic energy for wall collisions for the thermostat can be changed in the **simulation** menu under **options**. The default value is 10%. See Section XI for a discussion of adiabatic expansions and compressions.

E. The Advanced Area



⊞ Advanced

⊟ Advanced

Frame Rate 30.6fps

☐ Write averages to file

☐ Write positions to file

☐ Exhaustive RDF calculations

The **advanced** area contains a box that displays the frame rate and check boxes to initiate periodic output of averages and/or of cumulative distributions or particle positions. The advanced area is hidden by default because it is not for normal use. Click the + sign by **Advanced** to expand it.

Frame Rate Display

The **Frame Rate** display shows the rate of graphics display in frames per second. The goal is to display 30 frames per second. The displayed number should stay between about 27 and 33 or the motion will appear jerky and uneven. If the number is consistently below 30 and/or fluctuates widely, your graphics card and CPU are not able to do the required computation adequately. If so, you should try using fewer particles. You can also try turning off **lighting** under the **view** menu.

Write Averages to File (Periodic Output of Averages)

Click the **Write averages to file** box to call up a menu that will allow you to write out the desired averages to a file on your hard disk. The menu allows you to choose which averages you want saved, the name and location of the file, and the time interval between outputs. If you choose a short time interval, you will write a large file and take up a large space on your hard disk. Choosing a large interval will write a smaller file, but will lose some information about shorter time fluctuations. The file is written in *.csv (comma separated value) format and can be imported easily into Excel or other spreadsheet programs for analysis. Output of this information begins when the OK box is clicked. Output of this information stops when the **Write Averages to File** box is unchecked or when another change in the system is made.

Output of Cumulative Distributions to a File

Clicking the box labeled **Write Averages to File** will also, using the same menu, allow you to choose which cumulative distributions you wish to write out, as well as the name and location of the file. The file is written in *.csv (comma separated value) format and can be imported into Excel or other spreadsheet programs for analysis. The cumulative distributions begin to accumulate when the OK box is clicked, and the cumulative distributions are written out when the Write Averages to File box is unchecked or when another change is made to the system.

Write Positions to File (Output of Coordinates)

Click the **Write positions to file** box to call up a menu that will allow you to write out the coordinates and momenta of each particle to a file on your hard disk. The menu allows you to choose the name and location of the file and the time interval between outputs. If you choose a short time interval, you will write a large file and take up a large space on your hard disk. Choosing a large interval will write a smaller file, but will lose some information about shorter time fluctuations. The file is written in *.csv (comma separated value) format and can be imported easily into Excel or other spreadsheet programs for analysis. Output of this information begins when the OK box is clicked. Output of this information stops when the **Write positions to File** box is unchecked or when another change in the system is made. You can use the information stored in this file to investigate correlation functions or other details of the simulation.

Exhaustive RDF calculations

You should check this box if you want accurate numerical values for the radial distribution function (RDF) when you print out RDF distributions. Checking this box forces Boltzmann 3D to sample interparticle distances at time intervals equal to half of the predicted collision time (or other choice set in **simulation options**). This calculation slows down the calculation of other quantities, especially at high **simulation speeds**, so leave this box unchecked unless written RDF data is your interest.

Activation Energy Diagram

When **Reaction Mode** is checked in the **Simulation** menu, an energy diagram appears in the **Advanced Area** that controls the activation energies for forward and reverse reactions and displays the number of red (reactant) and blue (product) particles (see Simulation Menu, Reaction Mode, Section I.I). Prediction for the equilibrium average number of product particles is displayed. See Section X for an explanation of these predictions.

F. The Program Control Area

At the bottom left of the window are the **Clock**, the **Simulation speed** control, the **Zoom** control, the **Piston Level** control, and four buttons used to **Restart, Pause**, **Advance,** and **Reverse** the simulation.

Clock

The **clock** displays the time elapsed since the simulation began. Because atomic motion is so incredibly fast, this clock runs extremely slow. The simulation clock only advances about 1 picosecond (ps) in one second of time on your wall clock or wrist watch. The exact conversion will depend on how many particles you have in the arena, the kind of computer you use, and the **Simulation speed** control setting you choose.

Simulation Speed Control

The **Simulation Speed** control is used to speed up the simulation to allow the system to reach equilibrium faster, to accumulate statistics rapidly, or to slow down the simulation to see smoother, more detailed motion. Click the up or down arrows next to the simulation speed text box. The default value is 1x and each successive step is twice the speed of the previous one up to 2048x. You should choose a setting that makes motion appear the best on your computer. Use high settings to accumulate cumulative averages quickly when you don't care to follow the actual motion. On high settings, the particles move so far between frames that one can no longer follow motion from one frame to the next. Alternatively, you can turn off the graphical display of particle movement altogether by simply choosing the **display** option under the **view** menu.

Zoom Control

Use the **Zoom** control to adjust the magnification of the arena so it appears best in the screen resolution you use. The default magnification is 1 pixel equals 1 nm. Click on the up arrow of the **Zoom** control to increase the magnification of the arena shown on the screen. The magnification increases in steps of about 25% each time the arrow is clicked. If another increase in magnification would exceed the number of pixels of screen resolution, the instruction is ignored. Click on the down arrow of the **Zoom** control to decrease the magnification of the arena. The minimum allowed is 33%. Type a number in the box to zoom to a desired percent if you want a zoom factor in between or outside the preset steps.

Restart Button

Each time you click on the **Restart** button, **Boltzmann 3D** starts a new simulation using the number, radius, mass, and temperature values currently displayed in the **particle information** area. Positions and directions for the particles are assigned randomly. All particles are red and are on the left side of a divided arena. If you want to add blue particles of a different mass or radius, you must make the necessary changes in the **mass** and **radius** labels before changing the number value.

Pause Button

Click on the **Pause** button to pause the simulation. You may make changes, examine distributions, etc. while paused. Click on the same button, which now shows a **Play** symbol, to resume the simulation.

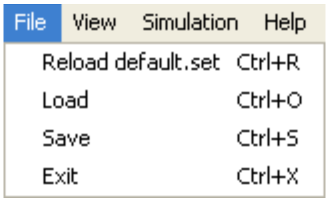
Advance Button

Click on the **Advance** button to move the simulation to the next frame of motion. It is only available while the simulation is paused.

Reverse Button

Click on the **Reverse** button to reverse the direction of all the particles. The clock will continue to move in the forward direction, but the motion will retrace the previous motion backwards (as well as it can be done).

G. The File Menu



Loading Saved Values (Load Menu Item)

Click on **Load** to display the standard open dialog box. Only files with the extension *.set may be loaded and will be shown in the list. Click on the file you want to load and then click open. This loads the chosen *.set file with saved attributes. Some example demonstration files are provided with **Boltzmann 3D** and described in Section X and used in Suggestions for Exploring Kinetic Theory (Section II). Others may be generated using the **Save** menu item. A saved file named default.set will be loaded when the program is started. You can save files with your personal preferences in this file, as well as in any of the saved files.

Saving Initial Values (Save Menu Item)

Click on **Save** to create special sets of conditions for classroom demonstrations, student assignments, or experiments. Then to re-create the simulation, load the appropriate initial value file. To create a file of initial values, click on the **Save** menu item, and choose the **complete** or **basic** option. The **basic** option stores the following quantities to the file: temperature, number, radius, and mass of both red and blue particles; distribution maxima and minima (all of them); selected boundary mode; piston parameters, which distribution is shown first; color scheme; lighting type; dimension (and 3D viewing angle); hole diameter; arena size; pressure averaging time; and forget time. Positions and velocities are then assigned randomly upon loading. If the arena is divided, red particles will appear on the left and blue on the right when loaded. The **complete** option will also save the individual particle position and velocities to enable the same simulation to be observed repeatedly. Use the **complete** option to begin a simulation with red and blue particles both on the left of a divided arena. After choosing an option for saving, enter the name of the file (the *.set extension is automatically appended) on the save dialog box and click the save box.

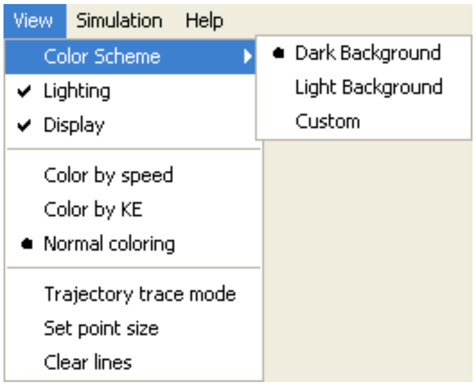
Exit

Click on **Exit** under the **File** menu to exit the program.

H. The View Menu

Color Scheme

Click on **Color Scheme** to change the appearance of **Boltzmann 3D**. Click on **Dark Background** for a 2D simulation with black arena background. Click on **Light Background** for a 2D simulation with white arena background and gray walls. In both background choices, the balls will be red for the first set of particles and blue for the second set. Default colors in 3D are gold for right and front walls and neutral beige for the others. Click on **Custom** to choose your own colors for background, walls, divider, and particles.



Lighting

Check **Lighting** to toggle shading of the particles on and off in 2D simulations. With shading off, you will have higher contrast and motion may be faster and smoother, particularly if shading is done by your CPU rather than your video card. This option is not active in 3D because it lacks depth perception.

Display

Check **Display** to toggle the drawing of motion in the arena on and off. You can turn off display of motion if you want to accumulate averages or cumulative distributions without watching the associated motion.

Color by speed and Color by KE

Check **Color by speed** to color each particle according to its speed. Check **Color by KE** to color each particle according to its kinetic energy. Particles with high values will be red; particles with low values will be blue; and particles with intermediate values will be green, yellow, orange, etc. When the **Instantaneous Speed Distribution** is chosen under **Color by speed**, the histogram bars will be colored with the same colors as the particles in the corresponding speed ranges. When the **Kinetic Energy Distribution** is chosen under **Color By KE**, the histogram bars will be colored with the same colors as the particles in the corresponding kinetic energy ranges. These colors can be controlled by resetting the maximum and minimum of the plot. Any particle with a value lower than the minimum will be dark blue. Any particle with a value higher than the maximum will be dark red.

Normal coloring

The default colors are red for red particles and blue for blue particles, although these colors can be reset using **Color Scheme** in the **View Menu**.

Trajectory trace mode

Click this option to have **Boltzmann 3D** trace out the path of the center of each particle rather than draw the instantaneous position of the particle as it moves. The path is indicated by a square dot for each frame. The color of the dot is the same as the color of the particle. The distance between trajectory points is proportional to the speed of the particle.

Set point size

Click this option to enter the size of dots drawn in **Trajectory trace mode**. The default is 2 pixels. Any integer number from 1 to 20 is allowed.

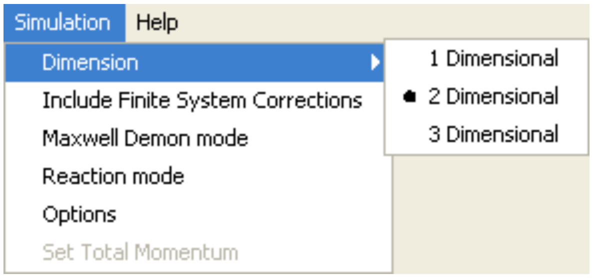
Clear lines

Click this option to clear all points drawn in **Trajectory trace mode**.

I. The Simulation Menu

Dimension

Click on **Dimension** to choose a 1D simulation that places particles horizontally across the arena, a 2D simulation that allows particles to move both horizontally and vertically in the visible arena, or a 3D simulation with perspective and shading to show depth. Arena size for 1D or 2D simulations can be adjusted by dragging the lower-right corner of the **Boltzmann 3D** window using the mouse. Arena size in all three dimensions can be set using the **options** menu tab under the **simulation** menu.



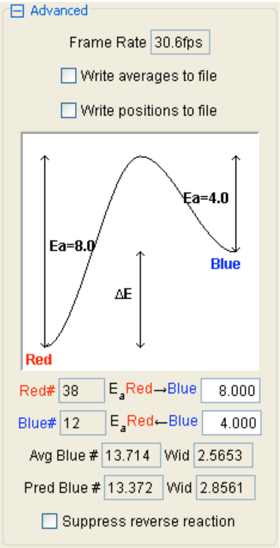
Include Finite System Corrections

Check **Include Finite System Corrections** to calculate predictions for pressure and plotted distributions, averages, and widths using any known corrections for a finite system. The default mode is to calculate predictions based on an infinite system. (See Section IX).

Maxwell Demon Mode

Use this option to pretend to be a Maxwell demon, a fictitious microscopic creature imagined by James Clerk Maxwell to decrease the entropy of a system by opening a hole in a wall to allow fast particles to escape but closing it to prevent slow particles from escaping. When clicked with a **Divided Box with Walls** or **Divided Box with a Hole** boundary mode, the hole may be opened or closed repeatedly by clicking either button. This option is the closest thing to a video game offered by **Boltzmann 3D**. Starting with particles on the left side, see if you can arrange a system with particles at an average energy on the right hand side three or more times that in the left hand side. It requires patience and quick mouse work. Hint: use **Color by Speed** or **Color by KE** so you can tell which particles have high energy. If the edge of a particle overlaps the wall as it closes, that particle is allowed to move as if the hole had not closed.

Reaction Mode



Frame Rate 30.6fps

☐ Write averages to file

☐ Write positions to file

Red# 38 E_a Red→Blue 8.000

Blue# 12 E_a Red←Blue 4.000

Avg Blue # 13.714 Wid 2.5653

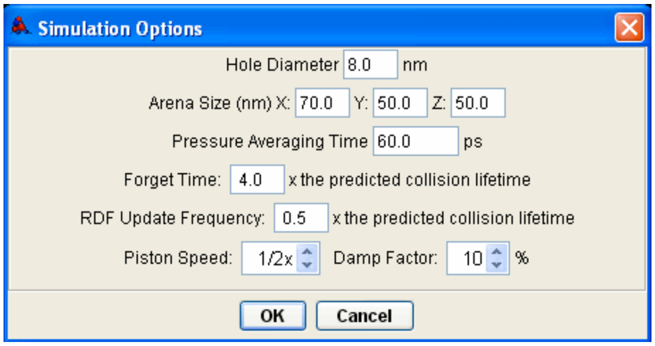
Pred Blue # 13.372 Wid 2.8561

☐ Suppress reverse reaction

This mode opens a diagram in the **Advanced Area** where you can enter two values, the activation energy for reaction from red to blue and the activation energy for reaction from blue to red. The difference, E_{ar} – E_{ab}= ΔE, is the potential energy difference for a reaction that converts to red particles to blue particles. The default values are E_{ar} = 8 kJ/mol and E_{ab} = 4 kJ/mol. If two red particles collide with relative kinetic energy projected along the line connecting their centers greater than E_{ar}, they change to two blue particles of the same size, and their kinetic energy is reduced by ΔE. If two blue particles collide with their relative kinetic energy projected along the line connecting their centers greater than E_{ab}, they change to two red particles, and their kinetic energy is increased by ΔE. This mode can be used to study reaction kinetics of bimolecular reactions or equilibrium. Activation energy values can be set by the user. The current numbers of red and blue particles are displayed below the energy diagram as well as the time averaged number of blue particles and the predicted equilibrium number and the time averaged width of the distribution of blue particles and predicted values. When reaction mode is invoked, the default plot changes to **particle number** to show the number of red and blue particles as a function of time. Other plots can be accessed as usual by selecting from the drop-down menu. Because the amount of kinetic energy changes when particles react, the value displayed in the temperature box in the particle information area will not indicate the instantaneous temperature in reaction mode after the simulation starts.

Options

Use **Options** to set several simulation parameters. These include hole diameter for a divided arena, arena size, averaging time for wall pressure calculations, forget time, RDF update frequency, and piston parameters.



Hole Diameter 8.0 nm

Arena Size (nm) X: 70.0 Y: 50.0 Z: 50.0

Pressure Averaging Time 60.0 ps

Forget Time: 4.0 x the predicted collision lifetime

RDF Update Frequency: 0.5 x the predicted collision lifetime

Piston Speed: 1/2x Damp Factor: 10 %

Horizontal and vertical arena size may also be adjusted by dragging the lower right corner using the mouse. The **forget time** option controls the equilibration time before cumulative distributions are set to zero to allow non-equilibrium initial conditions to be *forgotten*. The default value is 4 times the predicted collision time. It can be set by the user to any multiple of the predicted collision time. When this time is reached in any simulation, the cumulative distributions for all quantities are cleared, and accumulation of data begins anew. The RDF update frequency is the time interval between sampling of interparticle distances. The default value is half the predicted collision lifetime, but it can be set to any desired multiple. You can also set the speed of piston motion and the maximum percent change (damping factor) of kinetic energy in a wall collision taking place during isothermal compressions or expansions.

Set Total Momentum

When using periodic boundary conditions, use the **Set Total Momentum** option to adjust the total momentum in 1D, 2D or 3D. After random assignment of particle velocity, the total momentum of the particles is, by default, set to zero. Zeroing the total momentum is accomplished by shifting and rescaling individual particle momenta while preserving the total kinetic energy. However, the user can set the total momentum to a non-zero value to study a system with an overall drift rate. You can set the total momentum to have a non-zero value in any coordinate direction you wish, but many of the predictions assume a frame of reference that is not drifting.

J. The Help Menu

About

The **About** menu option displays information about **Boltzmann 3D, including** the **Boltzmann 3D** version number and the Java and OpenGL graphics libraries it uses.

Help

The **Help** menu option brings up the help file for **Boltzmann 3D** in your default browser program**,** which includes much of this document. While using the **Help** menu, the current simulation will continue to run.

Quick Start

The **Quick Start** menu option brings up the *Quick Start and Troubleshooting Guide* for **Boltzmann 3D**, a brief series of questions and pointers for those using **Boltzmann 3D** the first time.

Update

The **Update** menu option opens your default browser to the **Boltzmann 3D** website to allow you to check for newer versions of **Boltzmann 3D**.

Manual

This link connects to the full **Boltzmann 3D** manual (this document) in pdf format, including theory, formulas, and suggestions for **Boltzmann 3D** in education and research.

II. Suggestions for Exploring Kinetic Theory

Each one of the following suggestions could be used as an in-class demonstration, a hands-on laboratory assignment, or any other type of exercise.

A. Distribution of Molecular Velocities

Start a simulation with the **Instantaneous Speed** plot visible. Note that all speeds are the same to start. While you watch, the histogram broadens out to the shape of the Maxwell-Boltzmann distribution shown by the black line. Compare the velocity distribution for particles of different mass and at different temperatures. Compare the difference between 2D and 3D motion. Sample file examples: 3Dhyd100.set, 3Dhyd1000.set, 3Doxy100.set, 3Doxy1000.set, 2Dhyd100.set, 2Dhyd1000.set, 2Doxy100.set, 2Doxy1000.set

Questions for exploration:

1. Does the **Instantaneous Speed** histogram look more like the Maxwell-Boltzmann distribution if you have more particles in the simulation? Why or why not? (Compare with 30 vs. 480 particles.)

2. How long does it take for the **Instantaneous Speed** distribution to become like the Maxwell-Boltzmann distribution? Try changing the temperature, the number of particles, mass and/or the radius to see how these variables change your answer. Is the time required just a multiple of the average number of collisions? (To find the time per collision, look at the **collision time** plot.)

3. How does the average speed change with temperature? With mass? With radius? With number of particles? Molecular kinetic theory predicts that the average speed should be proportional to the square root of temperature, inversely proportional to the square root of mass, and unaffected by radius or number of particles. Try some changes and see if the predictions of molecular kinetic theory (predicted numbers and orange lines) agree with the actual simulation (average numbers and green lines). Sample file: 3Dmixture.set

4. The RMS speed is always higher than the average speed (about 25% in 1D, about 13% in 2D, and about 8.5% in 3D). Can you explain why these two quantities are different? (Formulas are given in Section IV, but can you explain *why*?)

B. Boyle's Law

How does the pressure change when you change the volume of a gas while keeping the temperature constant? Start a 2D simulation with about 50 particles of radius 0.5 nm with T = 300 K. Note the wall pressure. You can set the pressure averaging time to a large value like 200 ps to suppress fluctuations (use the **options** item under the **simulation** menu). Then resize the arena by dragging with the mouse on the lower-right corner. Has the pressure changed?

A second method to study Boyle's law is to use an isothermal compression using the **Piston level** control. Note the pressure before and after the compression. Compute PA (PV in 3D or PL in 1D) before and after. Are they the same?

Questions for exploration:

1. Can you explain *why* the pressure increases when the volume decreases? (Hint: Watch the frequency of wall collisions.)

2. The 2D analogue of Boyle's Law is that *PA* should be constant (at constant *T*). Using the pressures you noted and the area of the arena from the length and heights read from the axis labels (or the more accurate values found under the **options** item of the **simulation** menu), do you get the same value for *PA* before and after resizing? If not, how big is the deviation?

3. Repeat the experiment using particles of different mass and radius but the same temperature. Does the pressure depend on the mass or size? Why or why not?

4. Perform isothermal compressions in small increments in piston mode. Record the pressure at several volume increments and check to see if *PA* is constant.

C. Amonton's Law

How does the pressure of a gas change when you change the temperature? Start a simulation with about 30 particles of radius 0.5 nm at 300 K in a divided arena. Now change the temperature to 1200 K and use the mouse to drop the same number of the same size particles in the right half of the arena. What are the left and right pressures? Sample file: 3Dhotcold.set

Questions for Exploration:

1. Why is the pressure higher for the hotter gas? (Hint: Compare the speed of the particles when they collide with the wall and the frequency of these collisions.)

2. Repeat the same experiment with blue particles of different mass or size (but not too big). Does your conclusion depend on mass or size?

3. Repeat the experiment using a number of different temperatures and plot the pressure versus temperature. At what temperature does the pressure extrapolate to zero pressure?

4. What does absolute zero (0 K) physically mean about the motion of particles?

D. Charles' Law

How does the volume of a gas depend on its temperature with the pressure held constant? Start a simulation of 50 to 100 particles of radius 0.5 nm (keep radius small to avoid non-ideality corrections) at 300 K and note the pressure. Now set the temperature to 150 K and click **Restart**. You will note that the pressure has decreased, in accordance with the conclusions of Amonton's Law (Section II.C above). Now, resize the arena using the boxes found under the **options** tab under the **simulation** menu by multiplying the length in one direction by ½. Is the pressure the same as the original pressure?

Questions for Exploration:

1. Repeat the experiment for several different temperatures and plot the volumes against temperature. Does the volume extrapolate to zero as T approaches 0 K?

2. Contrary to Charles' Law, volume does not go to zero because the volume of the particles is

not zero. To what volume does the extrapolation to 0 K approach?

E. Pressure Versus Number of Particles

Start a simulation with 40 to 50 particles of radius 0.5 nm and T = 300 K. Then click on the **Divided Box with Walls** icon so the red particles are on the left-hand side. Now click with the mouse on the empty right-hand side of the arena until you have just half the number of blue particles as red particles. Compare the pressure on the two sides. Sample file: 3Dpresnum.set

Questions for Exploration:

1. Is the pressure in the right side exactly half of what it is in the left side? If not, why not?

2. What is the physical reason pressure is proportional to the number of particles? (Hint: Compare the frequency of wall collisions.)

F. Deviations from the Ideal Gas Equation

Does the pressure predicted by the ideal gas equation agree with the pressure calculated in the simulation? If not, why not? The ideal gas equation is $PV = nRT$ in 3D, $PA = nRT$ in 2D and $PL = nRT$ in 1D. Here, n is the number of moles of gas, R is the gas constant, T is the absolute temperature (in Kelvin), V is the 3D volume, A is the 2D area, and L is the 1D arena length.
Questions for exploration:

1. How does the deviation from ideal gas behavior depend on the number of particles? On the radius of the particles? On the temperature? On the mass of the particles? (See Section V for some theoretical predictions.)

2. How would the pressure be affected if the particles attracted each other?

G. Effusion

Divide the arena by clicking the **Divided Box with Walls** icon. Then click **Restart**. Now all the balls are in the left side. Change the mass and radius to different values and drop some blue particles into the left side until you have about the same number of blue ones as red ones. Now click the **Divided Box with a Hole** icon. Watch and see if the heavier particles take longer to get out. One way to do this would be to wait until about one fourth of the lighter ones have escaped and compare one fourth with the fraction of heavier ones that have escaped. Sample files: 3Deffuse.set, 2Deffuse.set

Questions for exploration:

1. Using two separate experiments, measure the rate of effusion by timing how long it takes one fourth of the particles of two different masses that differ by a factor of four to escape. Does it take twice as long for the heavier particles? What would Graham's law predict?

2. Why can't you wait until *all* the particles have escaped in the first question?

H. Entropy Production

Using the mouse, place 24 particles of radius 1 nm all in the lower left corner of the arena while the temperature gauge reads 0 K (make sure temperature reads zero before you place each particle). The particles should remain motionless. Then set the temperature box to read 9999 K and drop one blue particle in the upper right of the arena and see what happens. This initial state is very ordered, with the positions of the first 24 particles in one small region and the kinetic energy all concentrated in the last particle. Because entropy is a measure of disorder, the initial state has low entropy. The laws of probability say that it is much more likely for the positions to be spread over the whole arena and the kinetic energy to be shared equally among all the particles. This would be a higher entropy state. Sample file: 3Dentropy.set, 2Dentropy.set

Questions for Exploration:

1. Does the simulation lead from the low entropy state to one of higher entropy? Why?

2. After the first collision, how long does it take to reach a more-or-less final entropy equilibrium with particles spread over the arena and a velocity distribution of 400 K?

3. Suppose you started your simulation with particles in random positions and with random velocities. How likely is it that you will come to a point where all 25 particles are in the lower left quarter of the arena? Can you estimate the probability?

I. Brownian Motion

To see a good approximation of the motion of a dust particle, for example, start a simulation with one large particle (about 200 amu) with radius 20 nm. Then change the radius indicator to 0.2 nm, mass indicator to 2 amu, and finally the number indicator to 200, to add many small blue particles. Sample files: 3Dbrownian.set, 2Dbrownian.set

Questions for exploration:

1. Does the red particle appear to stagger around due to random bombardment by small particles from all sides?

2. Does the motion of the red particle qualitatively change if it is only half as heavy (100 amu)?

J. Diffusion

Start a simulation with 50 to 100 red particles and add one more, identical particle (but blue) by clicking the mouse in the arena. Just sit back and watch the blue particle move. Try this in **trajectory trace** mode. Start a second simulation with a divided arena with light red particles on the left side and heavy blue particles on the right side. Then click on the undivided arena icon to remove the barrier. Sample files: 3Ddiffuse.set, 2Ddiffuse.set, 2Dliquid.set

Questions for Exploration:

1. Does the blue particle eventually wander around throughout the arena? Does it go fast sometimes and slow other times?

2. Is the velocity distribution of the blue particle as shown by its cumulative distribution the same as for the other particles? Is its average velocity the same as the other particles?

3. What about mean free path and collision rate?

4. In the second simulation, do the light particles diffuse into the right side faster than the heavy

particles diffuse into the left side? Why?

K. Heating and Cooling

Start a simulation with about 50 red particles with a temperature of 300 K. Check the RMS velocity and/or kinetic energy of these particles. Press **Pause** and add 20 or 30 identical blue particles with a temperature of 1200 K (this means they have a velocity twice as great as the RMS velocity of the other particles). Now press **Pause** again to continue. Watch the average velocity of the blue particles decrease as they cool to the average temperature of the red particles. Sample files: 3Dhotcold.set, 2Dhotcold.set

Questions for exploration:

1. How long does it take the blue particles to cool? Express your answer as a multiple of the mean collision time.

2. Does the temperature or average kinetic energy of the red particles change from its initial value as the blue particles cool? Why? Can you predict the final RMS average velocity of the red and blue particles? Are they the same?

3. Do your answers to the first two questions change if you start with a divided arena with the red particles on the left and the blue ones on the right and then remove the division?

L. Independence of Kinetic Energy Distribution on Mass and/or Radius

Start a simulation with about 30 to 50 large, heavy particles (e.g. m = 100 amu and r = 0.8 nm), then add 30 to 50 small, light blue particles (e.g. m = 10 amu, r = 0.4 nm). After a while, compare the velocity distributions of the two sets of particles by clicking back and forth between the **Red** and **Blue** buttons. You should find that the heavy particles have a lower average velocity than the light particles. Then, compare the kinetic energy distributions of the two sets of particle. Except for statistical fluctuations, the two sets should have the same average and the same distribution. Sample files: 3Dmixture.set, 2Dmixture.set

Questions for exploration:

1. Is the average kinetic energy proportional to temperature? If so, what is the proportionality constant? Does the proportionality constant depend on dimension?

2. Try the same experiment changing the mass or the radius, but not both. Do you get the same results?

3. Can you guess *why* average kinetic energy should behave the way it does? (See Section IV for a brief explanation.)

M. Equilibrium

Click on **Reaction mode** in the **simulation mode** and look at the energy diagram in the **Advanced Area**. When two red particles collide with high relative velocity, they can change into blue particles with the loss of kinetic energy. When this happens, the temperature changes because kinetic energy decreases. Two blue particles can collide and change back to red particles and release the energy again. After a long time, equilibrium will develop between red and blue particles. Sample file: 2Dequil.set

Questions for exploration:

1. Should there be more blue particles or red particles at equilibrium if energy is absorbed when red particles change to blue particles? Why?

2. Increase ΔE by typing in a larger number for the activation energy from red to blue. Does that increase or decrease the average number of blue particles? Why? How does the ratio of blue particles to red particles depend on ΔE?

3. How does the average number of blue particles depend on the activation energy, E_a, for the forward and reverse direction without changing ΔE? [Theory predicts it should not, although the rate of approach to equilibrium does (see item P below)].

N. Adiabatic Compression

Click on the **Piston** icon and select the **adiabatic** button. Then move the slider to a desired final volume measured as a fraction of the initial volume. Now click **Move Piston** to start the compression. The piston does work on the system by transferring energy to colliding particles. The system is considered to be insulated from the surroundings, so the extra energy raises the temperature of the gas.

Questions for exploration:

1. Why does the temperature go up when you compress a gas? [Use the law of conservation of energy]

2. Can you predict the final temperature of the gas as a function of the final volume? [See Section XI for some ideas]

3. When you expand back to the initial volume in a second adiabatic expansion, why is the final temperature not the same as the initial temperature?

4. For a given final volume, does the final temperature depend on how fast the piston moves? [Try changing the piston travel time in the **simulation** menu?]

5. For a given final volume, does the final temperature depend on the number of particles?

O. Partial Pressure

Create a mixture of red and blue particles. You can choose the number, mass and radius of each. Click **both** and read the pressure. Now click **red** and read the pressure. Now click **blue** and read the pressure. How are these pressures related? Try this with different mixtures of red and blue particles.

Questions for exploration:

1. Is the total pressure the sum of the red partial pressure and the blue partial pressure? Why?
2. How does the computer know how much pressure is due to each color of particle if the pressure has to be averaged over a long time involving many wall collisions?

P. Bimolecular Kinetics

Start a simulation with 100 to 200 particles at 300 K. Choose reaction mode from the **simulation** menu. Use 10 kJ/mol for both E_a and E_b. Select suppress reverse reaction in the advanced area to only allow reaction from red to blue, not the reverse reaction from blue to red. The rate of a reaction is the reciprocal of the length of time it takes for a given number of product particles to form. Observe how long it takes for 10 particles (or some other fixed number) to turn blue when you change the number of particles, particle radius, particle mass, temperature, and the activation energy. Repeat each simulation several times and calculate an average.

Questions for exploration:

1. How much longer does it take for 10 blue particles to form if the starting number of particles 100 compared to starting started with 200 particles? Why? [Theory predicts a factor of four]
2. How much longer does it take for 10 blue particles to form if the radius is increased by a factor of 2? Why? [Theory predicts a factor of four in 3D and a factor of two in 2D]
3. How much longer does it take for 10 blue particles to form if the mass in increased by a factor of 2? Why? [Theory predicts a factor $2^{-1/2}$]
4. How much longer does it take for 10 blue particles to form if the temperature is increased by a factor of 2? Why? [Theory predicts the logarithm of the time should be a linear function of the reciprocal of the temperature with a slope proportional to E_a.]

III. Frequently Asked Questions Concerning Boltzmann 3D

Where did the name Boltzmann 3D come from?

Boltzmann 3D was named after the Austrian physicist Ludwig Boltzmann (1844–1906) who worked out much of the theory of entropy and statistical mechanics. The Boltzmann distribution for energy and the Maxwell-Boltzmann distribution for velocity both carry his name. Boltzmann also developed an equation, named in his honor, to describe the approach of a system to equilibrium. However, he is most famous for his definition of entropy in terms of the logarithm of the number of microstates. This equation is carved on his tomb and is written $S = k\ln W$, where S is entropy, W is the number of accessible microstates, and k is the proportionality constant, now called Boltzmann's constant, which depends on the units of entropy. A picture of Boltzmann's tomb is included in the program documentation courtesy of Tom Schneider (NIH). The universal gas constant, R, is just Avogadro's number times Boltzmann's constant. Thus, Boltzmann's constant is the gas constant per molecule.

On my computer, the particles move quite slowly. Can I speed them up so it looks snappier?
The best way to make the simulation run faster is to adjust the simulation speed control until it looks the best on your computer. You can also make the program appear faster by decreasing the number of particles, decreasing their mass, and increasing the temperature. Another way is to get a faster computer.

Because updating and redrawing the screen is quite computer intensive, another way to make the simulation run faster is to turn off the **display** option. This choice omits the display of particles on the screen. If you are interested in computing long-time averages rather than watching the corresponding motion, you can use this option. Even so, compared to the actual motion of particles, the simulation is slowed down by an incredible factor of something like 10^{12}.

I can't change the number of particles. What's wrong?
It may be that you have the red button depressed. Since any new particles added after a restart are blue, changing the number of red particles is not allowed. Click on either the **Both** or **Blue** button and add blue particles of the desired size and mass. Then, if you want them all to be red, hit restart. Alternatively, if the particles are densely packed, there may not be enough space for another randomly positioned particle. In that case, you can try enlarging the arena, using smaller particles, or try entering the particles at prescribed positions using a saved *.set file.

How do I report a bug or get customer support?
Phone or post inquiries to Randy Shirts (see cover page to this document) or email boltzhelp@chem.byu.edu. Most of the problems have occurred in downloading and getting the program to run. These errors are usually a result of incompatibilities between video cards and drivers and the Java language used in the program. See the troubleshooting guide for known fixes. We try our best to get around these differences but make no guarantees. Suggestions for improvement or added functionality, examples of saved demonstrations, examples of laboratory exercise worksheets, and stories of how you used **Boltzmann 3D** are all welcome. We want you to be satisfied—or your money back!

Using initial particle positions and velocities, can you predict what will happen a long time in the future?
No! Hard disk motion simulated in **Boltzmann 3D** has been mathematically proven to be chaotic. Even infinitesimal changes in the initial position of a particle will result in enormous effects after a few collisions. This sensitivity to initial conditions or to small changes is called *chaos*. For example, if we were able to calculate the paths of 100 particles with a radius of 0.4 nm and a mean free path of 8 nm exactly, and do two calculations side-by-side in two identical halves of the arena with one initial position changed by 10^{-10} nm on one side (less than 0.001 of the radius of the nucleus!) compared to the other side, the simulations would look the same for a period of time less than several multiples of the average collision lifetime. Slowly, differences would start to be noticeable until after approximately 9 or 10 collision lifetimes, they would be completely different. The differences between the two simulations grow exponentially because the convex particle edges magnify any difference in position when a collision occurs. Chaos

becomes noticeable sooner if you have more particles (because there are more collisions), but even with only a few particles, the motion is unpredictable after a small number of collisions and about 1 ns of simulation time. (See Section VI for a brief mathematical derivation of these concepts.) Sample file equal.set has 20 particles in each half of a 2D divided arena. The initial conditions are identical for each set, but differences in the round-off error between the computations for the two halves result in visibly different motions beginning by 90 ps of elapsed simulation time.

The arena is so small! Aren't quantum mechanical effects important for systems of this size?
Quantum effects become important for distances comparable to the deBroglie wavelength of the particle. This effect is biggest for light particles which are moving slowly. Even for particles of mass 1 amu moving at a typical speed at a temperature of 1K, the deBroglie wavelength is only one tenth of the length of the arena. This means that the typical kinetic energy of a particle in this extreme case is about 400 times the zero-point energy of the system. Even in this case, quantum effects would be small. For more typical particles at more typical temperatures, quantum effects are quite negligible.

How well does Boltzmann 3D conserve energy?
The internal calculations are done to about fourteen decimal places of accuracy. You can check the conservation of energy by simulating one type of molecule for a long time and looking at values of the RMS velocity or kinetic energy. In a typical test, the kinetic energy changes by a relative error of $\pm 5\text{x}10^{-14}$ in several microseconds of simulation time.

How are the heights of the histogram bars scaled?
The histogram bars are scaled so the maximum height of any bar is the full height of the graph field. The predicted distribution is also scaled so its maximum value in the included range is full height of the graph. This method is usually satisfactory.

What is the purpose of periodic boundary conditions in Boltzmann 3D?
Most research-level simulations of atomic-size particles use periodic boundary conditions. This means that when a particle leaves one edge of the arena, it reappears at the opposite edge with the same velocity. This method has several advantages in modeling bulk gas and liquid behavior. The primary advantage is that there are no walls, so no wall collisions take place. Consequently, the walls do not influence the properties of the gas. In a simulation with walls, for example, a large fraction of the particles are near at least one wall. This proximity can change some of the properties of the gas, especially at high density. Most beginning students can understand the motion of particles in a box simulation where particles impact with walls easier than in one where particles leave one side of the arena and appear at the other side. Pressure, as a property arising from wall collisions, is also easier to understand in a “box simulation.”

Can I study phase transitions like evaporation, melting, etc. using Boltzmann 3D?
No. Attractive forces, which are the primary cause of phase transitions, are not included in the motion of the particles in **Boltzmann 3D**. Java is too slow to include realistic attractive forces in a real-time visualization. Look for a future program written in C to visualize particles interacting with a Lennard-Jones potential.

How are the initial velocities chosen?

From your choice of mass, m, and temperature, T, all of the initial velocities are chosen as $v = (dRT/m)^{1/2}$, where d is the dimension. This way each particle has a kinetic energy, $mv^2/2$ equal to the average value of $dRT/2$. In fact, the definition of temperature is twice the average value of kinetic energy divided by dR. In other words, each dimension must have average kinetic energy $RT/2$ (a principle called *equipartition*). The initial direction of motion and position are determined randomly using a random number generator.

How did Boltzmann 3D come about?

In 1994, long before **Boltzmann 3D** was written, Dr. Shirts produced a two-dimensional DOS program called **MotionBYU** to help chemistry and physics students at BYU visualize the distribution of molecular velocities. **MotionBYU** was programmed in TurboPascal by David L. Summers, an undergraduate electrical engineering student with game programming experience, and may have been the first program to perform real-time simulations of particle motion on emerging Intel 486 computers that made such simulations possible for the first time. **MotionBYU** was rewritten in C++ by David Summers as a Windows program and renamed **Boltzmann. Boltzmann** was commercially distributed by Trinity Software until the current version, **Boltzmann 3D** was written. **Boltzmann 3D** was written by Scott R. Burt, an undergraduate chemistry major, and Benjamin J. Lemmon, an undergraduate computer science major, during the summer of 2004. The major improvements for **Boltzmann 3D** are that it is programmed in Java using OpenGL so as to be portable to other platforms besides Windows, the addition of 1D and 3D simulations, and an improved user interface. Some of the design for the user interface was developed by the Center for Instructional Design at Brigham Young University.

IV. Some Results from Molecular Kinetic Theory

This section is much more technical than previous sections. Here we derive the formulas for the predicted values. We also discuss how the results for a 2D system differ from those for a 3D system. If you are a beginning student and if you have not had calculus, the material in this section may be too difficult for you. If you are an advanced undergraduate student or more experienced, this section contains much of the elegance and beauty of kinetic theory. A more detailed account of these issues is found in almost any textbook on kinetic theory or statistical mechanics.

One-dimensional systems are, in some ways, easier than 2D or 3D systems, but they are in other ways harder. For this reason, 1D systems are discussed separately (see Section VIII).

The predicted distribution of atomic velocities is called the Maxwell-Boltzmann distribution named after James Clerk Maxwell and Ludwig Boltzmann. *Disorder* (or *entropy*) is the main concept needed to understand the distribution of molecular velocities. It would be very unlikely that all the particles in a sample would be going the same speed. A good analogy is flipping a coin. It is very unlikely (though possible) that you will get heads 100 times in a row. The most likely numbers are 50 heads and 50 tails, but you might also get 51 heads and 49 tails, etc. Assuming you have a fair coin, the mathematical probability of getting n heads in N flips (and N − n tails) is given by elementary combinatorics

$$P_N(n) = \frac{N!}{n!(N-n)!2^N}\,. \tag{1}$$

For $N = 100$ and $n = 50$, $P_{100}(50) = 0.07959$. So even though 50 heads is the most probable number, it is still not very probable (less than 8%). However, the probability of getting somewhere between 45 and 55 heads is about 73%. To be specific, there are 2^{100} different possible outcomes of 100 coin flips, but 73% of the possibilities have heads come up between 45 and 55 times. Thus, the probability distribution has a “most probable value” of 50 and a “width” of about 5 (more properly called a half-width at half-height). The reason 50 heads don’t come up very often is that there are so many other possibilities that are about equally likely. In like manner, there are many different combinations of velocities possible for N particles. There are still quite a few possible combinations even when we only consider those with exactly the same total kinetic energy. Ludwig Boltzmann showed (McQuarrie, ch. 2) that the probability of a particle having a particular energy, E in kJ/mol, was related to the temperature, T, by the Boltzmann distribution formula

$$P(E) \propto \exp\left(\frac{-E}{RT}\right), \tag{2}$$

where R is the gas constant. For coins, the probability was a function of the integer index n; however in Eq. (2), the probability is a function of the continuous index E. In Eq. (1), the sum of all the probabilities $P_N(n)$, $n = 0,1,2,...N$ can be shown to add up to exactly one. In Eq. (2), the *integral* must be one:

$$\int_0^\infty P(E)dE = 1\,. \tag{3}$$

Because, for hard-sphere particles where there is no potential energy, the energy is just the kinetic energy, $\frac{1}{2}mv^2$, we can replace E by the kinetic energy and get a distribution of molecular velocities. In one dimension (1D), this is done as follows:

$$f(v_x) = \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_x^2}{2RT}\right), \tag{4}$$

where we have determined the proportionality constant by

$$\int_{-\infty}^{\infty} dv_x f(v_x) = 1\,, \tag{5}$$

where the integral, called a Gaussian integral, can be looked up in many different tables of integrals (CRC *Handbook of Chemistry and Physics*; Gradshteyn and Ryzhik, *Table of Integrals Series and Products*; etc.). Note that the most probable value for v_x is 0. The average value of v_x is also zero (from integrating $v_x f(v_x)$ from $-\infty$ to ∞). However the average value of v_x^2 (from integrating v_x^2f(v_x) from $-\infty$ to ∞) is RT/M. Thus, the width, or more properly the root-mean-square deviation, of the 1D velocity distribution is $(RT/M)^{1/2}$.

The velocity distribution for two dimensions can be obtained from Eq. (4) by multiplying $f(v_x)$ by an identical function, f(v_y), and for three dimensions, by multiplying by two functions, f(v_y) and f(v_z). We can do this because the kinetic energy is separable into components and those components are independent. From this point, we will divide our discussion into parallel columns to show you how the mathematics works out in both two and three dimensions. In either case, we can get the total energy by adding up the kinetic energy in each of the vector components.

Two dimensions (2D)

$$\begin{aligned} f(v_x, v_y) &= \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_x^2}{2RT}\right) \\ &\times \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_y^2}{2RT}\right) \\ &= \left(\frac{M}{2\pi RT}\right) \exp\left(\frac{-M(v_x^2 + v_x^2)}{2RT}\right), \end{aligned} \tag{6}$$

Three dimensions (3D)

$$\begin{aligned} f(v_x, v_y, v_z) &= \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_x^2}{2RT}\right) \\ &\times \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_y^2}{2RT}\right) \\ &\times \left(\frac{M}{2\pi RT}\right)^{1/2} \exp\left(\frac{-Mv_z^2}{2RT}\right) \\ &= \left(\frac{M}{2\pi RT}\right)^{3/2} \exp\left(\frac{-M(v_x^2 + v_y^2 + v_z^2)}{2RT}\right), \end{aligned} \tag{7}$$

where the proportionality constant has been adjusted so that

$$\int_0^\infty dv_x \int_0^\infty dv_y f\left(v_x, v_y\right) = 1 \tag{8}$$

We can now express this probability function in terms of speed, the magnitude of the velocity, $v = (v_x^2 + v_y^2)^{1/2}$, by changing the variables of integration from v_x and v_y to v and the polar angle θ. After integrating θ from 0 to 2π, the result is the probability of molecular speed, v:

$$f(v) = \left(\frac{Mv}{RT}\right)\exp\left(\frac{-Mv^2}{2RT}\right). \tag{10}$$

where we got a factor of 2π from the angle integration and a factor of v from the area element for polar coordinates. The most probable speed is obtained by differentiating $f(v)$ with respect to v, setting the result equal to zero, and solving for $v_{max} = (RT/M)^{1/2}$. The average speed is obtained by multiplying $f(v)$ by v and integrating from 0 to ∞: $v_{avg} = (\pi RT/2M)^{1/2}$. The RMS speed is obtained by multiplying $f(v)$ by v^2 and integrating from 0 to ∞ to get the average of v^2, then taking the square root: $v_{rms} = (2RT/M)^{1/2}$. You will note that these three speeds are not equal: the RMS speed is slightly greater than the average speed which is slightly greater than the most probable speed (the peak in the $f(v)$ curve). We will define the width of the speed distribution to be $(v_{rms}^2 - v_{avg}^2)^{1/2}$. The width is easily computed to be $[(2-\pi/2)RT/M]^{1/2} \approx 0.655\ v_{max}$.

where the proportionality constant has been adjusted so that

$$\int_0^\infty dv_x \int_0^\infty dv_y \int_0^\infty dv_z f(v_x, v_y, v_z) = 1, \tag{9}$$

We can now express this probability function in terms of speed, the magnitude of the velocity, $v = (v_x^2 + v_y^2 + v_z^2)^{1/2}$ by changing the variables of integration from v_x, v_y, and v_z to v and the spherical angles θ and ϕ. After integrating θ from 0 to π and ϕ from 0 to 2π, the result is the probability of molecular speed, v:

$$f(v) = 4\pi\left(\frac{M}{2\pi RT}\right)^{3/2} v^2 \exp\left(\frac{-Mv^2}{2RT}\right). \tag{11}$$

where the factor of 4π comes from the angle integration and the factor of v^2 comes from the volume element for spherical coordinates. The most probable speed is obtained by differentiating $f(v)$ with respect to v, setting the result equal to zero, and solving for $v_{max} = (2RT/M)^{1/2}$. The average speed is obtained by multiplying $f(v)$ by v and integrating from 0 to ∞: $v_{avg} = (8RT/\pi M)^{1/2}$. The RMS speed is obtained by multiplying $f(v)$ by v^2 and integrating from 0 to ∞ to get the average of v^2, then taking the square root: $v_{rms} = (3RT/M)^{1/2}$. You will note that these three speeds are not equal: the RMS speed is slightly greater than the average speed which is slightly greater than the most probable speed (the peak in the $f(v)$ curve). We will define the width of the speed distribution to be $(v_{rms}^2 - v_{avg}^2)^{1/2}$. The width is easily computed to be $[(3-8/\pi)RT/M]^{1/2} \approx 0.476\ v_{max}$.

Another quantity of interest is the average *relative* velocity of particles involved in a collision. This quantity is obtained by taking two velocity distributions, $f(v_1)$ and $f(v_2)$ for two particles, then converting from individual velocities to relative and center-of-mass velocities (McQuarrie, pp. 365–371). When this is done, the relative velocity appears with the reduced mass, $\mu = M_1 M_2/(M_1 + M_2)$. If particles 1 and 2 have the same mass, then $\mu = M/2$. When this is expressed in $(v_{rel})_{avg}$, you need to replace M by $M/2$, and because M appears in the denominator under the square root sign for all the velocities above, $(v_{rel})_{avg} = \sqrt{2}\ \mathrm{v}_{avg}$. This result holds in 2D or 3D. We will now return to parallel columns and derive predicted formulas for the mean free path in 2D and 3D.

Two Dimensions

Consider a circular particle of radius, r, moving relative to other particles with velocity, v_{rel} in an arbitrary direction which we will call x. The fraction of particles which have a collision with another particle between x and $x + dx$ is given by dn/n, where n is the number of particles per unit area. Each of the N particles in the area, A, presents a target of width 4r, because a collision will occur if two particles approach closer than 2r on either side. Thus, the fraction of particles undergoing a collision from x to $x + dx$ is:

$$\frac{1}{n}\frac{dn}{dx} = -4rn_0, \qquad (12)$$

which has the unique solution $n(x) = n_0 \exp(-4rn_0x)$. Here $n(x)$ is the number of particles that have not had a collision since some reference location, and n_0 is the total number of particles per unit area. The average distance traveled is given by:

$$\lambda = -\frac{1}{n_0}\int_0^\infty x\frac{dn}{dx}dx = \frac{A}{4rN}. \qquad (14)$$

Three Dimensions

Consider a circular particle of radius, r, moving relative to other particles with velocity, v_{rel} in an arbitrary direction which we will call x. The fraction of particles which have a collision with another particle between x and $x + dx$ is given by dn/n, where n is the number of particles per unit volume. Each of the N particles in the volume, V, presents a target of area $\pi(2r)^2$, because a collision will occur if two particles approach closer than 2r on any side. Thus, the fraction of particles undergoing a collision from x to $x + dx$ is:

$$\frac{1}{n}\frac{dn}{dx} = -4\pi r^2 n_0, \qquad (13)$$

which has the unique solution $n(x) = n_0 \exp(-4\pi r^2 n_0 x)$. Here $n(x)$ is the number of particles that have not had a collision since some reference location, and n_0 is the total number of particles per unit volume. The average distance traveled is given by:

$$\lambda = -\frac{1}{n_0}\int_0^\infty x\frac{dn}{dx}dx = \frac{V}{4\pi r^2 N}. \qquad (15)$$

To be consistent, we have to divide both of these results by $2^{1/2}$. You many consider this to be because both particles which collide are moving and the distance traveled will be shorter because v_{rel} is larger than v_{avg} by a factor of $2^{1/2}$. The mean free path is predicted to be $\lambda = A/(4\ 2^{1/2}\ r\ N)$ in 2D and $\lambda = V/(4\ 2^{1/2}\ \pi\ r^2\ N)$ in 3D, where N is the number of particles of radius r, A is the area of the arena, and V is the volume of the sample. Now we will derive an expression for the time between collisions. The method is very much like that for the path.

Two Dimensions

If the fraction of particles which have a collision with another particle between *t* and *t* + *dt* is *dn*/*n*, where *n* is the number of particles per unit area, then:

$$\frac{1}{n}\frac{dn}{dt} = -4rn_0 v_{rel}, \tag{16}$$

which has solution $n(\mathrm{t}) = n_0\exp(-4rn_0v_{rel}t)$. Here $n(t)$ is the number of particles per unit area that have not had a collision since some reference time, and n_0 is the total number of particles per unit area. The average time between collisions is given by:

$$\tau = -\frac{1}{n_0}\int_0^\infty t\frac{dn}{dt}dt = \frac{A}{4rNv_{rel}}. \tag{18}$$

Now replacing v_{rel} by $2^{1/2}v_{avg}$, the mean collision time is predicted to be $\tau = A/(4\ 2^{1/2}\ r\ N\ v_{avg})$, where *N* is the number of particles, and *A* is the area of the arena.

Three Dimensions

If the fraction of particles which have a collision with another particle between *t* and *t* + *dt* is *dn*/*n*, where n is the number of particles per unit volume, then:

$$\frac{1}{n}\frac{dn}{dt} = -4\pi r^2 n_0 v_{rel}, \tag{17}$$

which has solution $n(t) = n_0\exp(-4\pi r^2 n_0 v_{rel}t)$. Here $n(t)$ is the number of particles per unit volume that have not had a collision since some reference time, and n_0 is the total number of particles per unit volume. The average time between collisions is given by:

$$\tau = -\frac{1}{n_0}\int_0^\infty t\frac{dn}{dt}dt = \frac{V}{4\pi r^2 Nv_{rel}}. \tag{19}$$

Now replacing v_{rel} by $2^{1/2}v_{avg}$, the mean collision time is predicted to be $\tau = V/(4\ 2^{1/2}\ \pi r^2\ N\ v_{avg})$, where *N* is the number of particles, and *V* is the volume of the arena.

In either two or three dimensions, the collision rate, the number of collisions per unit time, can easily be seen to be τ^{-1}. Using the definitions of $n(x)$ and $n(t)$ above, we can now substitute in for λ and τ to obtain probability distributions for path length and collision time. In each case, we require that the integral of the distribution from 0 to ∞ be equal to one.

$$P(x) = \frac{1}{\lambda}\exp\left(\frac{x}{\lambda}\right) \quad ; \quad P(t) = \frac{1}{\tau}\exp\left(\frac{t}{\tau}\right). \tag{20}$$

From either of these exponential distributions, it is a simple calculus problem to show that the average path or lifetime is given by:

$$< x >= \int_0^\infty \frac{xdx}{\lambda}\exp\left(\frac{x}{\lambda}\right) = \lambda \quad ; \quad < t >= \int_0^\infty \frac{tdt}{\tau}\exp\left(\frac{t}{\tau}\right) = \tau\,. \tag{21}$$

Likewise you can show that

$$< x^2 >= \int_0^\infty \frac{x^2dx}{\lambda}\exp\left(\frac{x}{\lambda}\right) = 2\lambda^2 \quad ; \quad < t^2 >= \int_0^\infty \frac{t^2dt}{\tau}\exp\left(\frac{t}{\tau}\right) = 2\tau^2\,. \tag{22}$$

Thus, forming the width or RMS deviation, $\sigma_x = (2\lambda^2 - \lambda^2)^{1/2} = \lambda$, and $\sigma_t = (2\tau^2 - \tau^2)^{1/2} = \tau$. In general, for any exponential distribution, the average is equal to the width.

One additional probability distribution is used in **Boltzmann 3D**: the distribution of number of collisions each particle has experienced since the start of the simulation. If we assume that each collision is uncorrelated with the previous collision (the technical name for this is a Poisson process), and if the probability distribution for each collision is given by Eq. (20), then the fraction of particles having experienced n collisions in time t is [McQuarrie, p. 525]:

$$f(n,t) = \frac{1}{n!}\left(\frac{t}{\tau}\right)^{n} \exp\left(-\frac{t}{\tau}\right). \tag{23}$$

This distribution is normalized so the sum of $f(n,t)$ from $n = 0$ to $n = \infty$ is one. This formula gets very difficult to evaluate for large values of n, so we use an approximation. We also want to express it as a function of the collision rate, $R = n/t$. We expect the average collision rate to be τ^{-1}. Using Stirling's approximation for $n!$ and differentiating with respect to n, we obtain the value of n for which $f(n,t)$ is a maximum as $n \approx t/\tau$. Then, expanding Eq. (23) about the maximum in terms of the variable $R = n/t$, we obtain:

$$f(R) = \left(\frac{t\tau}{2\pi}\right)^{1/2} \exp\left(\frac{1}{2}(R - \frac{1}{\tau})^{2} t\tau\right). \tag{24}$$

This corresponds to a bell-shaped curved centered at $R = \tau^{-1}$ (as expected) with a width of $(t\,\tau)^{-1/2}$. Thus, when the rate of collisions for each particle is histogrammed, the average should be τ^{-1} and the width should decrease as time increases. The bell-shaped curve of Eq. (24) is plotted by **Boltzmann 3D** when the collision rate distribution is plotted; however, it is not continually redrawn as a function of time. Instead, the curve is redrawn whenever the width changes by 22%—quite often at the beginning of a simulation, but less often as the simulation time, *t*, increases.

For normal reactions at constant temperature and pressure, the equilibrium between red and blue particles in reaction mode would be given by an equilibrium constant $K = N_b/N_r = \exp(-\Delta G/RT)$. However, Boltzmann 3D does not do simulations at constant temperature and pressure, but at constant number of particles and energy. For this situation, the equilibrium constant (assuming an infinite system is given simply by $K = N_b/N_r = \exp(-\Delta E/RT)$.

Now let us summarize the quantities that we use in **Boltzmann 3D** and the differences between 2D and 3D in the following table:

Quantity	**2D value**	**3D value**
Most probable speed	$v_{max} = (RT/M)^{1/2}$	$v_{max} = (2RT/M)^{1/2}$
RMS velocity	$v_{rms} = (2RT/M)^{1/2}$	$v_{rms} = (3RT/M)^{1/2}$
Average velocity	$v_{avg} = (\pi RT/2M)^{1/2}$	$v_{avg} = (8RT/\pi M)^{1/2}$
Average relative velocity	$v_{rel} = 2^{1/2} v_{avg}$	$v_{rel} = 2^{1/2} v_{avg}$
velocity width	$\sigma_v = (2-\pi/2)^{1/2}(RT/M)^{1/2}$	$\sigma_v = (3-8/\pi)^{1/2}(RT/M)^{1/2}$
avg velocity component	$v_x = v_y = 0$	$v_x = v_y = v_z = 0$
velocity component width	$\sigma_{vx} = \sigma_{vy} = (RT/M)^{1/2}$	$\sigma_{vx} = \sigma_{vy} = \sigma_{vz} = (RT/M)^{1/2}$
mean free path	$\lambda = A/(4\ 2^{1/2} rN)$	$\lambda = V/(4\ 2^{1/2} \pi r^2 N)$
free path distribution width	$\sigma_\lambda = \lambda$	$\sigma_\lambda = \lambda$
collision lifetime	$\tau = A/(4\ 2^{1/2} rNv_{avg})$	$\tau = V/(4\ 2^{1/2} \pi r^2 N v_{avg})$
collision lifetime width	$\sigma_\tau = \tau$	$\sigma_\tau = \tau$
collision rate	$R = 1/\tau$	$R = 1/\tau$
collision rate width	$\sigma_R = (t\tau)^{-1/2}$	$\sigma_R = (t\tau)^{-1/2}$
equilibrium constant	$K = N_b/N_r = \exp(-\Delta E/RT)$	$K = N_b/N_r = \exp(-\Delta E/RT)$

Adiabatic Expansions: When the piston compresses a gas, it imparts additional momentum to each particle that collides with it. This additional momentum increases the kinetic energy and, thus, the temperature of the gas. You can understand this on a macroscopic scale because the piston must do work on the gas in counteracting the pressure of the gas. This work increases the energy of the gas.

When a gas expands, it does work on against the external pressure. This work is taken directly from the energy of the gas contained in its motion, lowering the temperature. On a microscopic level, a piston moving away from a colliding particle takes energy away from the particle, much like a recoiling bat decreases the reflecting speed of a fastball in a bunt.

V. Non-Ideal Gases: The Virial Expansion

It may be a surprise to you that the simple, circular disks with no attractive forces between them that we call particles in **Boltzmann 3D** do not obey the ideal gas equation. The ideal gas equation is accurate to the extent that two assumptions are obeyed: 1) attractive forces between particles are negligible, and 2) the volume (or area in 2D, length in 1D) occupied by the particles is negligible compared to the size of the container. The first assumption is obeyed exactly by the particles in **Boltzmann 3D**. The forces between the particles are identically zero until the particles collide. In a collision, the repulsive forces are instantaneously infinite, but this does not cause any deviations from nonideality. The second assumption is not obeyed by the particles in **Boltzmann 3D** because they have finite size. The van der Waals equation of state (found in most general chemistry textbooks) for such a system would be:

$$P(V - Nb) = nRT, \tag{25}$$

where b is a constant which is related to the volume of a particle ($4\pi r^3/3$ in 3D, πr^2 in 2D). We can estimate b by noting that circular disks are touching each other (close-packing) when $A = (2\ 3^{1/2}/\pi)\ \pi r^2 N$. This means that if b were approximately $1.102\pi r^2$, the pressure would become infinite as the volume approached the close-packing volume. It turns out that this equation results in corrections for nonideality that are too small by a factor of about two. More rigorous theory must be used to produce more accurate corrections.

The traditional way to express deviations from nonideality is with the virial expansion in terms of the density, ρ (N/A in 2D or N/V in 3D) given by the following expression (McQuarrie, p. 163):

$$\begin{aligned} \frac{PV_d}{nRT} &= 1 + B_2\rho + B_3\rho^2 + B_4\rho^3 + \ldots \\ &= 1 + x + b_3x^2 + b_4x^3 + \ldots \end{aligned} \tag{26}$$

where V_d is the d-dimensional volume of the container (A in 2D and V in 3D). In Eq. (26), B_n is called the n^{th} virial coefficient, $x = B_2\rho = B_2N/V_d$, and $b_n = B_n/B_2^{\,n-1}$. A rigorous calculation of the second virial coefficient for hard disks gives $B_2 = 2\pi r^2$ in 2D, and $B_2 = 16\pi r^3/3$ in 3D. And b_3 is approximately 0.782004 in 2D and exactly 0.625 in 3D. Numerical simulations beginning in 1959 (see references to B. J. Alder and coworkers) and theoretical derivations (Clisby, 2003) have resulted in an expression for the virial expansion (the right-hand side of Eq. 26) for hard disks and hard spheres that is accurate to eight virial coefficients and approximate after that:

$$\frac{PV_d}{nRT} = 1 + x\left(\frac{1 + e_1x + e_2x^2 + e_3x^3}{1 + f_1x + f_2x^2 + f_3x^3}\right) = 1 + xQ(x). \tag{27}$$

where the constants have been adjusted to agree with the known virial coefficients and are given in the table below.

	2D	3D
e_1	−0.25233539	1.30542183
e_2	0.02704323	0.11627913

e_3	0.00077137	0.02452331
f_1	−1.03433982	0.68042183
f_2	0.30366975	−0.59593402
f_3	−0.01974721	0.09148337

Boltzmann 3D uses Eq. (27) for pressure predictions when the **Real** button is selected. In addition, corrections for nonideality for the mean free path, mean collision time, and collision rate are expressed in terms of the ratio of polynomials within the large parentheses in Eq. (27) ($Q(x)$):

$$\begin{aligned} \lambda_{real} &= \lambda_{ideal} \div Q(x) \\ \tau_{real} &= \tau_{ideal} \div Q(x) \\ Rate_{real} &= Rate_{ideal} \times Q(x). \end{aligned} \tag{28}$$

Boltzmann 3D uses Eq. (28) for the indicated predictions when the **Real** button is selected.

In addition to these corrections, which are necessary for high density in bulk fluid, additional corrections are needed for a finite system. For example, the definition of x used in Eq. (27) with walls, to a first approximation, should be changed from B_2N/V_d to $B_2(N-1)/V_d$ because a particle can collide with only $N-1$ *other* particles. This change is needed for a box with walls, but is cancelled out for periodic boundary conditions due to an effect coming from the conservation of center-of-mass momentum. For a simulation with walls, V_d in the density should be replaced by $(L_x - 2r)(L_y - 2r)$ in 2D and $(L_x - 2r)(L_y - 2r)(L_z - 2r)$ in 3D to correct for the fact that the center of a particle cannot get closer to a wall than a distance r.

In addition, when a particle is near an edge or corner, it blocks other particles from regions that are normally accessible. This effect leads to corrections to the second virial coefficient. In 2D, the effective second virial coefficient is, where σ is twice the radius,

$$B_2^{eff} = B_2\left[1 - \frac{2}{\pi}\left(\frac{\sigma}{L_x} + \frac{\sigma}{L_y}\right) + \frac{6}{\pi}\left(\frac{\sigma}{L_x}\frac{\sigma}{L_y}\right)\right] \tag{29}$$

In 3D, the effective second virial coefficient is

$$B_2^{eff} = B_2\left[1 - \frac{1}{2}\left(\frac{\sigma}{L_x} + \frac{\sigma}{L_y} + \frac{\sigma}{L_z}\right) + \frac{11}{20\pi}\left(\frac{\sigma}{L_x}\frac{\sigma}{L_y} + \frac{\sigma}{L_x}\frac{\sigma}{L_z} + \frac{\sigma}{L_y}\frac{\sigma}{L_z}\right) + \frac{15}{2\pi}\left(\frac{\sigma}{L_x}\frac{\sigma}{L_y}\frac{\sigma}{L_z}\right)\right] \tag{30}$$

Similar corrections to the third and higher virial coefficients are still under investigation.

VI. Hard Spheres: A Chaotic System

Hard-disk and hard-sphere systems have been extensively studied mathematically. The dynamics is sufficiently simple to be amenable to exhaustive analysis. The most important result of this analysis is that hard disk and hard sphere dynamics is *mixing*. *Mixing* is a technical mathematical term. A mixing system is both ergodic and chaotic. An ergodic system is one where time averages are equal to ensemble averages. (We take this to mean that the time average for one particle is equal to the distribution average among all particles.) A chaotic system is one where very small perturbations in the initial positions of particles result in large changes in the motions of the particles at later times; the effect of a perturbation grows exponentially in time. This extreme sensitivity to initial conditions is the hallmark of chaos. Physically, this sensitivity comes from the imaginary hard edge of the particles that we are simulating. Because they have a convex edge, small differences in the position of impact of two collision partners will be magnified in their resultant final directions. Any such difference will be magnified again and again by subsequent collisions, thus leading to the exponential growth of differences that is the essence of chaos. Let us demonstrate this result mathematically. Consider two particles which are moving toward a collision. If you project the path of one particle relative to the other as if they were point particles that would not collide, the distance of closest approach is called the impact parameter, b. A collision occurs if $b < 2r$. The final angle of deflection (in the center-of-mass frame of reference) is easily shown to be (McQuarrie, p. 370):

$$\theta(b) = \begin{Bmatrix} 2\arccos\left(\dfrac{b}{2r}\right) & \text{if} & b < 2r \\ 0 & \text{if} & b \geq 2r \end{Bmatrix} \tag{31}$$

This function decreases from 180° for b = 0 to 0° for $b = 2r$. The sensitivity of θ to b is given by $|d\theta/db|$ which increases from $(2r)^{-1}$ at $b = 0$ to ∞ at $b = 2r$. We will use the minimum value to make a conservative estimate. Thus if the initial impact parameter is perturbed by an amount δb_1, the first deflection angle is perturbed by an amount $\delta\theta_1 = \delta b_1/(2r)$. The second impact parameter is perturbed by $\delta b_2 = \delta\theta_1 \ell_1$ where ℓ_1 is the distance traveled. Now we can compute the perturbation in the second deflection angle due to δb_2 the same way we did $\delta\theta_1$: $\delta\theta_2 = \delta b_2/(2r) = (\ell_1/2r)\,\delta\theta_1$. Finally, we can estimate the general perturbation by replacing each of the distance values, ℓ_i, $i = 1, 2, 3, \ldots n$, by their average, the mean free path, λ, to obtain:

$$\delta\theta_{n+1} = \left(\frac{\lambda}{2r}\right)^n \frac{\delta b_1}{2r}, \tag{32}$$

which gives the perturbation in deflection angle after n+1 collisions due to a perturbation in the first impact parameter. This perturbation gets arbitrarily large as long as the mean free path is greater than $2r$ (actually, the system may still be chaotic for smaller λ values, but we are making a *conservative* estimate). Because angle values in these formulas are in radians (1 radian ≈ 57°), the path of a particle will be largely unknown when $\delta\theta = 1$. We can estimate the number of collisions needed to lose predictive ability by setting the left-hand side of Eq. (32) equal to 1, taking the logarithm of both sides and solving for n:

$$n = \frac{\ln\left(\frac{2r}{\delta b_1}\right)}{\ln\left(\frac{\lambda}{2r}\right)}. \qquad (33)$$

Choosing numbers appropriate to a simulation in **Boltzmann 3D**, you can estimate the number of collisions and the time needed to lose predictive ability from the initial conditions. Try using $\delta b_1 = 10^{-11}$ nm (about 10^{-5} times the radius of the nucleus and about the round-off error of our numerical calculations). The result should surprise even those who fully understand the mathematics! For example, with 50 2D particles of radius 1 nm in a 50 nm by 70 nm arena at 300 K, you can only predict the motion accurately through about 15 collisions per particle, or about 120 ps.

VII. Mixtures of Particles

This section gives you the formulas that are used to predict the averages when two different types of particles are being simulated. The results can be derived using the same methods used in Section IV; only the results are given here. Type 1 particles are red on the computer screen; type 2 particles are blue. We give only the predictions for type 1 particles. The predictions for type 2 particles may be obtained by interchanging the 1 and 2 subscripts. We give below only those quantities which depend on having more than one type of particle. For example, the quantities related to velocity averages are independent of the existence of other types of particles.

Quantity	**2D value**	**3D value**
mean free path	$\lambda_1 = A/\{2^{3/2}[2r_1N_1+(r_1+r_2)N_2]\}$	$\lambda_1 = V/\{2^{1/2}\pi[4r_1^2N_1+(r_1+r_2)^2N_2]\}$
free path distribution width	$\sigma_{\lambda 1} = \lambda_1$	$\sigma_{\lambda 1} = \lambda_1$
collision lifetime	$\tau_1 = A/\{2v_{avg,1}[2^{3/2}r_1N_1+ (r_1+r_2)N_2(1+m_1/m_2)^{1/2}]\}$	$\tau_1 = V/[\pi v_{avg,1}(2^{5/2}r_1^2N_1+ (r_1+r_2)^2N_2(1+m_1/m_2)^{1/2}]\}$
collision lifetime width	$\sigma_{\tau 1} = \tau_1$	$\sigma_{\tau 1} = \tau_1$
collision rate	$R_1 = 1/\tau_1$	$R_1 = 1/\tau_1$
collision rate width	$\sigma_{R1} = (t\tau_1)^{-1/2}$	$\sigma_{R1} = (t\tau_1)^{-1/2}$

When the **Both** button is selected, predicted averages shown are number-weighted averages of the predicted values for the two types of particles. The above predictions work well for an ideal gas. The corrections discussed in Section V for a real gas are less accurate when there are two types of particles.

VIII. One-Dimensional Systems

One-dimensional systems are different than 2D and 3D systems in that the particles cannot exchange places; they always stay in their initial order. Thus, particle i always has collisions with particle $i - 1$ on its left and particle $i + 1$ on its right, and it has no collisions with any other particles. If two particles with identical masses collide, the only way to conserve both kinetic energy and momentum in a collision is for them to simply exchange velocities. For hard spheres, this exchange takes place instantaneously. One of the consequences of this one-dimensional fact is that a system of N particles of the same mass will not relax from some initial velocity or energy distribution into the Maxwell-Boltzmann distribution. Instead, the velocity distribution remains the same as the initial distribution. Particles will exchange their velocities in a collision, but there will always be a particle with each of the initial velocities, except for sign changes because of wall collisions. However, if as few as just one particle of a second mass is in the system, the Maxwell-Boltzmann distribution correctly predicts the time-averaged behavior.

For the same reason that 1D particles stay in their initial order, particles cannot move with equal probability over the whole width of the arena. The i^{th} particle is limited to the region between $x = 0 + r_i + 2\sum_{j=1}^{i-1} r_j$ and $x = L - r_i - 2\sum_{j=i+1}^{N} r_j$. In addition, in this region the probability for finding particle i increases to a maximum from a value of zero at each of these limiting points. Some of the space of length L is taken up by the particles (each particle has diameter $2r_i$), so the net available space (or free space) for motion is $L_{\text{net}} = L - 2\sum_{j=1}^{N} r_j$.

The velocity distribution for 1D particles is given by Eq. 4. The distribution for speed, $v = |v_x|$, is given by

$$f(v) = \left(\frac{2M}{\pi RT}\right)^{1/2} \exp\left(\frac{-Mv^2}{2RT}\right), \tag{34}$$

where $f(v)$ is normalized so

$$\int_0^\infty dv f(v) = 1 \tag{35}$$

The most probable velocity is $v = 0$, as is the average of v_x; however, the average speed is obtained by integrating $vf(v)$ from zero to ∞ and is $(2RT/\pi M)^{1/2}$. The RMS velocity is obtained by integrating $v^2 f(v)$ from zero to ∞ and is $(RT/M)^{1/2}$. Thus, the width of the distribution is v_{rms}.

The average relative velocity for 1D particles is $2^{1/2} v_{avg}$ just as in 2D and 3D. The mean free path for 1D particles is just the reciprocal of the probability per unit length of finding a particle, L/N. Once again, we need to correct for the relative motion by dividing by $2^{1/2}$ so $\lambda = L2^{-1/2}/N$. For the same reason, the mean time between collisions is $\tau = L2^{-1/2}/Nv_{avg}$. In both of these formulas, to correct for the space taken up by the particles themselves, L should be replaced by L_{net}.

The equation of state for N particles with identical masses and reflecting boundary conditions (walls) can be calculated exactly. The 1D pressure is the momentum transfer at a wall per unit time. When particles collide, they exchange velocities, so the collisions with the right wall can be distinguished by N different positive values of v_i.

$$P = \frac{\Delta p}{\Delta t} = \sum_{i=1}^{N} \frac{2\, m_i\, v_i}{\frac{2\, L_{net}}{v_i}} = \frac{2\, E_{tot}}{L_{net}} = \frac{NkT}{L_{net}} = \frac{NkT}{L - 2 \sum_{i}^{N} r_i} \qquad (36)$$

In the second equality, we note that each time there is a collision with the right wall, there is a momentum change from $+m_i v_i$ to $-m_i v_i$ (the magnitude of the change is $2m_i v_i$) and the time between each collision of this type is $2L_{net}/v_i$ because the motion has to go down the container and back again, an effective distance of $2L_{net}$ at velocity v_i. When a collision occurs, the momentum of type i is transferred to the next particle instantaneously and shifted a distance $|r_i + r_{i\pm1}|$ down the line. To get Eq. 34, next note that $m_i v_i^2/2$ is the kinetic energy of the i^{th} type, add them all up to get E_{tot}, and then use the definition of temperature, where in 1D, $2E_{tot} = 2NE_{avg} = NkT$. Note that the pressure becomes infinite when the particles take up all the space in the container (close packing). In this case, particle 1 and N are hitting the walls infinitely rapidly because momentum is being transferred down the box and returns instantaneously because L_{net} decreases to zero.

IX. Finite System Effects

The predictions derived above are technically only accurate for a system with an infinite number of particles. The systems studied using **Boltzmann 3D** have a fixed, finite number of particles and a fixed total energy and volume (N, V, E fixed). This has been called a *microcanonical system* or *microcanonical ensemble*. The velocity and energy distribution in the microcanonical ensemble are slightly different than given by the Boltzmann distribution (Lado, 1981; Roman, et al. 1996). These give rise to predictions that are slightly different than for the Maxwell-Boltzmann distribution. However, the differences between the standard result and that for a microcanonical system are always inversely proportional to the size of the system. Thus, with 100 particles you might expect differences of only about 1%. However, with only 20 particles, these differences may be as much as 5%. In addition, in periodic boundary conditions, the constraint that the center-of-mass momentum be conserved means that only $N - 1$ particles are independent. Thus, the velocity distribution for N particles with periodic boundary conditions has the same form as the distribution of $N - 1$ particles with the same average energy. The consequences of these differences in real, nanoscale systems and in computer simulations is the subject of current research. Some of these results are found in Shirts, et al., 2006.

X. Reaction Mode Equilibrium Constant

This section explains how **Boltzmann 3D** predicts the number of red particles and blue particles in reaction mode. In reaction mode, the user can set the activation energy for the forward reaction, E_{af}, and the activation energy for the reverse reaction, E_{ar}. The difference, $E_{af} - E_{ar} = \Delta E$, is the energy change for the reaction. **Boltzmann 3D** uses all these energies in kJ/mol. Only ΔE affects the equilibrium number of particles of each color. The separate values of E_{af} and E_{ar} will affect how fast equilibrium is reached, but not the equilibrium numbers.

In the process of motion, any time two red particles collide with sufficient kinetic energy that the relative kinetic energy projected along the line connecting their centers exceeds E_{af}, they change into blue particles, and their relative kinetic is changed by $-\Delta E$, that is, if ΔE is negative, their relative kinetic energy along the line of centers is increased. Similarly, any time two blue particles collide with sufficient kinetic energy that the relative kinetic energy projected along the line connecting their centers exceeds E_{ar}, they change into red particles, and their relative kinetic is changed by ΔE.

The equilibrium constant for the system involves the concentration of product (blue) particles divided by the concentration of reactant (red) particles. However, since there is only one reactant species and one product species and they are both contained within the same volume, the equilibrium constant is most easily expressed as the ratio of numbers N_1, the number of red reactant particles and N_2, the number of blue product particles. The total number of particles is always $N_1 + N_2 = N$.

When $\Delta E > 0$ in reaction mode, red reaction particles are more likely than blue product particles because it takes energy to make blue particles. For similar reasons, when $\Delta E < 0$, blue product particles are more likely than red reactant particles. This tendency is quantified by the equilibrium constant. For the chemical reaction written

$$2\,R \leftrightarrows 2\,B \tag{37}$$

The equilibrium constant is written

$$K = \frac{N_2^2}{N_1^2} = \frac{n_2^2}{n_1^2} \tag{38}$$

The same value is obtained when using the number of moles of each kind, n_1 and n_2. For normal chemical reactions taking place at constant temperature and pressure and involving many particles, students learn that

$$K = e^{-\Delta G^0/RT} \qquad \text{or} \qquad \ln K = \frac{-\Delta G^0}{RT} \tag{39}$$

where ΔG^0 is the standard change in Gibbs free energy for the reaction. In **Boltzmann3D**, rather than temperature and pressure, the constant variables are total energy and volume. For this case, the equilibrium constant is not given by Eq. (39). This section will derive the correct form of the equilibrium constant for large systems as well as some corrections to it for finite systems. However, under these conditions

$$K = e^{-\Delta E^0/RT} \qquad \text{or} \qquad \ln K = \frac{-\Delta E^0}{RT} \tag{40}$$

It should be noted that T in this last expression is not the initial temperature but the final temperature given by:

$$E = \tfrac{1}{2} dNkT = E_0 - \tfrac{1}{2}\langle N_2 \rangle \Delta E \tag{41}$$

Where E_0 is the total kinetic energy at the beginning of the simulation, d is the dimension of the simulation (1, 2 or 3), k is Boltzmann's constant (the gas constant R divided by Avogadro's number), and $\langle N_2 \rangle$ is the average number of product particles. The last expression has a two dividing the change in energy because ΔE results from the changing of a *pair* of blue particles. If ΔE is negative, the final temperature will be higher than the initial temperature because of extra kinetic energy deposited in the system, and the more blue particles formed, the higher the final temperature will be. Similarly, if ΔE is positive, the final temperature will be lower than the initial temperature. Since the final temperature cannot be known without knowing $\langle N_1 \rangle$ and $\langle N_2 \rangle$, we must use another method to calculate the equilibrium constant. The average number of blue product particles in reaction mode, $\langle N_2 \rangle$, can be found by doing a long simulation and taking the time average of N_2. Taking the time average $\langle N_2^2 \rangle$ allows you to calculate the root-mean-square deviation, too. As the simulation propagates in time, the instantaneous number of blue particles will fluctuate as particles react in the forward and reverse direction at random times. So the values of N_1 and N_2 in Eq. (38) are really averages of quantities that fluctuate and may not be integers.

Assuming the time average over a long time is the same as the ensemble average (ergodic hypothesis), the average number of blue particles and its root-mean-square deviation, σ, can also be calculated using the principles of statistical mechanics. The standard and most powerful way to calculate averages or probabilities in statistical mechanics uses the partition function. The partition function is the sum of all the different possible probability factors. These individual probability factors do not have to be normalized (probabilities add up to one), because they can be normalized by dividing by the sum. In this case, the probability of having N_1 red particles and N_2 blue particles is

$$C_1 \frac{N!}{N_1! N_2!} \left[E - \tfrac{1}{2} N_2 \Delta E \right]^{dN/2-1}, \tag{42}$$

where the exclamation point indicates the factorial function, d is the dimension, and C_1 is a constant dependent on N, particle masses and size, volume, dimension and Planck's constant, but independent of N_2 (see Shirts et al., 2006). The quotient factor in Eq. (42) is called a binomial coefficient and comes from combinatorial analysis and represents the number of distinct ways of choosing N_2 particles from N total particles.

The last factor is related to the surface of a dN dimensional hypersphere and is called the phase volume. For hard spheres, the kinetic energy of the system is the total energy and can be written as

$$E = \sum_{i=1}^{dN} \frac{p_i^2}{2m_i} \tag{43}$$

Thus, if the particles have identical masses, m, the quantity $2mE$ is the square of the radius of a sphere of dimension Nd, and the components of the individual momenta are the coordinates whose sum of squares is the radius squared. Thus, the points on the surface of a dN dimensional hypersphere represent the different ways kinetic energy can be distributed among the d different components of the momenta of the N different particles. This surface area is given by

$$S_{dN} = \frac{2\pi^{dN/2}(2mE)^{dN/2-1}}{\Gamma(dN/2)} \tag{44}$$

where $\Gamma(x)$ is the gamma function defined below in Eq. (47). This result was known at least by 1909 by Josiah W. Gibbs, after whom the Gibbs free energy is named. Since the number and mass of particles does not change as particles react in **Boltzmann 3D**, the only term that is important in this formula is $E^{dN/2-1}$. All the other terms can be factored out and taken into the overall normalization. To use in the partition function, we need only replace E by the kinetic energy $E - \frac{1}{2}N_2\Delta E$, where $N_2/2$ is the number of pairs of product particles formed, each of which subtracts ΔE from the kinetic energy. The partition function for this system is, therefore, given by

$$W = \sum_{N_2=0,even}^{N} \frac{C_1 N!}{N_1!N_2!}\left[E - \tfrac{1}{2}N_2\Delta E\right]^{dN/2-1} \tag{45}$$

where the sum includes only even values of N_2 because products particles are only created in pairs. Nor does the sum include any values of N_2 for which the quantity in brackets is negative, for example when ΔE is positive and there is not enough energy to create more than a maximum number of product particles. This sum could also be used if there were an odd number of blue particles to begin the simulation, only in that case the sum would only include odd values of N_2.

Note that when $\Delta E = 0$, the maximum probability occurs when $N_1 = N_2$ because of the maximum of the binomial coefficient factor and because the phase volume factor is then independent of N_2 and can be factored out of the sum. When ΔE is nonzero, the maximum shifts because the phase volume factor makes more volume available to whichever kind of particle has the lowest energy and thus the highest kinetic energy.

Given the partition function, the average of any quantity, in particular the *j*-th power of N_2, is given by

$$\left\langle N_2^j \right\rangle = W^{-1} \sum_{N_2=0,even}^{N} C_1 N_2^j \frac{N!}{N_1!N_2!}\left[E - \tfrac{1}{2}N_2\Delta E\right]^{dN/2-1} \tag{46}$$

Note the factors C_1 and $N!$ in the numerator and in W are independent of N_2, factor out outside the sum and cancel, so they are not really needed. Eq. (46) allows calculation of $\langle N_2 \rangle$ and $\langle N_2^2 \rangle$, and from these the width, or root-mean-square deviation, $\sigma^2 = \langle N_2^2 \rangle - \langle N_2 \rangle^2$. The precision of these calculations is limited only by the accuracy of the computer processor doing the calculation. The above describes how **Boltzmann3D** calculates the predicted values for reaction mode. Note that these results will give non-integer values for $\langle N_2 \rangle$ and $\langle N_2^2 \rangle$ which experience shows will agree with time averaged values from long classical mechanical simulations in **Boltzmann3D**. Note also that averages in angle brackets always represent ensemble averages over all the states with constant E (microcanonical ensemble).

Each term in the sum in Eq. (45) contributes to W. The most likely N_2 value will be the one with the largest contribution to the sum, the one with the largest number of possible microstates consistent with it. In other words, because all possible microstates are equally likely, the most likely value of N_2 occurs when it is associated with the largest number of microstates. If $|\Delta E|$ is small enough that the maximum of Eq. (42) is far enough away from $N_2 = 0$ and $N_2 = N$, it is a good approximation to treat the probability function as a Gaussian or normal distribution centered at the maximum with a width of σ. This way, W and averaged can be expressed simply in terms of the maximum value and width. Thus, we are looking for the value of N_2 making the maximum contribution to the partition function sum.

Calculus teaches us to find the maximum value of a function by finding zeros of its derivative. To differentiate the function of N_2 given by Eq. (43), we need to make two slight adjustments. First, the factorial function is a function of an integer argument. To differentiate it, we need an equivalent function of a continuous argument. The proper generalization is the gamma function

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t}\, dt \tag{47}$$

It is easy to show that, for integers, $\Gamma(n+1) = n!$. The second adjustment is that, rather than find the maximum of Eq. (42), we will find the maximum of its logarithm (proportional to the entropy). This is justified because

$$\frac{d \ln f(x)}{dx} = \frac{f'(x)}{f(x)} \tag{48}$$

Thus, as long as $f(x) \neq 0$, the zeros of $\ln f(x)$ are the same as those of $f(x)$. The maximum of the logarithm of Eq. (42) is the solution of the equation

$$-\psi(N_2+1) + \psi(N-N_2+1) - \frac{\Delta E}{2}\left(\frac{dN}{2} - 1\right)\left[E - \tfrac{1}{2} N_2 \Delta E\right]^{-1} = 0 \tag{49}$$

where the psi or digamma function, $\psi(z)$, is defined by $\psi(z) = \Gamma'(z)/\Gamma(z)$. We can simplify this using the asymptotic expansion of $\psi(z)$ for large arguments

$$\psi(z) = \ln(z) - \frac{1}{2z} - \frac{1}{12z^2} + \frac{1}{120z^4} - \frac{1}{252z^6} + \frac{1}{240z^8} - \ldots \tag{50}$$

and the exact recursion relation $\psi(z+1) = \psi(z) + z^{-1}$. Substituting these in Eq. (45) and neglecting terms inversely proportional to N_1 and N_2, we get

$$\ln K = 2\ln\left(\frac{N_2}{N_1}\right) = \frac{-\Delta E\left(\frac{dN}{2} - 1\right)}{E - \frac{1}{2} N_2 \Delta E} \tag{51}$$

In Eq. (51) the denominator of the right-hand side is just the kinetic energy of the system with $N_2/2$ pairs of blue particles, whose average is related to the equilibrium temperature. The equipartition theorem for kinetic energy states that (see Eq. (41) above)

$$\left\langle E - \tfrac{1}{2}N_2\Delta E \right\rangle = \frac{dNkT_e}{2} \tag{52}$$

Thus replacing the denominator of the right-hand side by its microcanonical ensemble average and neglecting a term inversely proportional to N, we obtain

$$\ln K = \frac{-\Delta E}{kT} \tag{53}$$

where T is the *equilibrium* temperature, not the initial temperature. If we replace ΔE by the value for a mole of product pairs and k by R, we obtain the analog of Eq. (3) for a system with constant E and V

$$\ln K = \frac{-\Delta E^0}{RT} \tag{54}$$

Note that Eq. (53) gives non-integer values for $\langle N_2 \rangle$ which are in excellent agreement with the statistical mechanical averages described above and time-averaged values from simulations. We are also assuming that the most probable value for N_2 gives the same value as the average value. This will be true if the maximum value is several multiples of σ away from $N_2 = 0$ and $N_2 = N$.

Equation (53) is exact in the large system limit ($N >>1$). We now seek corrections of order N^{-1} that are expected in such statistical mechanical results. It is useful to define a continuous variable x by

$$N_2 = \frac{N}{2}\left(1 + \frac{x}{4}\right) \qquad \text{and} \qquad N_1 = \frac{N}{2}\left(1 - \frac{x}{4}\right) \tag{55}$$

where this definition of x is chosen so $N_1 + N_2 = N$ and, to leading order, $x \approx \ln K$. We will seek corrections to the approximate relations above in terms of x.

We now introduce two competing definitions of entropy that have appeared in the scientific literature for a century. The earliest and most common definition, that of Boltzmann and Planck, is $S_B = k\ln(W\delta E)$, where W is the number of microstates of the system with energy E, δE is a range of energy much smaller than E, and k is Boltzmann's constant, R/N_A. The second definition is that of Hertz and Schlüter, in which $S_H = k\ln\Omega$, where Ω is the number of states with energy less than or equal to E. The surface of constant energy in dN dimensions is a hypersphere (or hyperellipsoid in the case of unequal masses). The Boltzmann-Planck (BP) entropy is the volume of a thin shell of thickness δE at the surface of that hypersphere. The Hertz-Schlüter (HS) entropy is the volume enclosed by the hypersphere. The two definitions can be shown to differ by terms of order N^{-1}. Thus, they agree for large systems. We will investigate the size of the corrections for small systems.

A comparison between the two entropy definition is done by rearranging Eq. (49)

$$2\left[\psi(N_2+1)-\psi(N-N_2+1)\right]=-s\Delta E\left[E-\tfrac{1}{2}N_2\Delta E\right]^{-1} \tag{56}$$

where the power to which kinetic energy is raised is $s=\frac{dN}{2}-1$ for BP entropy and $s=\frac{dN}{2}$ for HS entropy. We can obtain the relation between them by examining the following formula for a quantity proportional to the volume of the constant energy hypersphere.

$$\Omega=\sum_{N_2=0,even}^{N} C_2\frac{N!}{N_1!N_2!}\left[E-\tfrac{1}{2}N_2\Delta E\right]^{dN/2} \tag{57}$$

Now note that

$$\left\langle E-\tfrac{1}{2}N_2\Delta E\right\rangle=W^{-1}\sum_{N_2=0,even}^{N}\frac{C_1N!}{N_1!N_2!}\left[E-\tfrac{1}{2}N_2\Delta E\right]^{dN/2}=\frac{\Omega C_1}{WC_2} \tag{58}$$

Note also that

$$\frac{d\Omega}{dE}=\frac{dNC_2}{2C_1}W(E) \tag{59}$$

Thus, if we choose $dNC_2=2C_1$, Ω will have the proper behavior such that the number of states of microstates at energy E is the derivative of the number of microstates less than or equal to E. In other words, the surface area of the hypersphere is the derivative of the volume of the hypersphere. In this case, Eq. (58) reads

$$\left\langle E-\tfrac{1}{2}N_2\Delta E\right\rangle=W^{-1}\sum_{N_2=0,even}^{N}\frac{C_1N!}{N_1!N_2!}\left[E-\tfrac{1}{2}N_2\Delta E\right]^{dN/2}=\frac{dN\Omega}{2W} \tag{60}$$

We now introduce the thermodynamic temperatures appropriate for the two different definitions of entropy. For the BP entropy,

$$\frac{1}{T_B}=\left(\frac{\partial S_B}{\partial E}\right)_{N,V}=k\left(\frac{\partial\ln W}{\partial E}\right)_{N,V}=\frac{k}{W}\left(\frac{\partial W}{\partial E}\right)_{N,V}=k\left(\frac{dN}{2}-1\right)\left\langle\left(E-\tfrac{1}{2}N_2\Delta E\right)^{-1}\right\rangle \tag{61}$$

where T_B is the temperature defined by the BP entropy, S_B. Solving for the equivalent of Eq. (52)

$$\left(\frac{dN}{2}-1\right)kT_B=\frac{1}{\left\langle\left(E-\tfrac{1}{2}N_2\Delta E\right)^{-1}\right\rangle} \tag{62}$$

For the HS entropy, on the other hand,

$$\frac{1}{T_H} = \left(\frac{\partial S_H}{\partial E}\right)_{N,V} = k\left(\frac{\partial \ln \Omega}{\partial E}\right)_{N,V} = \frac{k}{\Omega}\left(\frac{\partial \Omega}{\partial E}\right)_{N,V} = \frac{kW}{\Omega} = \frac{kdN}{2\left\langle E - \frac{1}{2}N_2\Delta E\right\rangle} \tag{63}$$

where T_H is the temperature defined by the HS entropy, S_H. Solving for the equivalent of Eq. (52)

$$\frac{dNkT_H}{2} = \left\langle E - \tfrac{1}{2}N_2\Delta E\right\rangle = E - \tfrac{1}{2}\left\langle N_2\right\rangle\Delta E \tag{64}$$

Notice that the HS entropy agrees with the equipartition theorem, which is known to be exact for systems of hard spheres. For each entropy case, just as we did in Eq. (51), we replace the right-hand side of Eq. (56) with an appropriate microcanonical ensemble average. For the maximum of the BP entropy, we average the entire right-hand side

$$2\left[\psi(N_2+1) - \psi(N-N_2+1)\right] = -\frac{\Delta E}{kT_B} \tag{65}$$

For the maximum of the HS entropy, we average the denominator

$$2\left[\psi(N_2+1) - \psi(N-N_2+1)\right] = -\frac{\Delta E}{kT_H} \tag{66}$$

Thus, implementing the two different definitions of entropy involve solving the same equation, but the right-hand side of the equation has a different definition of temperature describing it.

The two temperatures are related by dividing Eq. (64) by Eq. (62) to obtain

$$T_H = T_B\left(1 - \frac{2}{dN}\right)\left\langle E - \tfrac{1}{2}N_2\Delta E\right\rangle\left\langle (E - \tfrac{1}{2}N_2\Delta E)^{-1}\right\rangle = T_B\left(1 - \frac{2}{dN}\right)\left(1 + \frac{\Delta E^2}{4d^2Nk^2T_e^2} + \cdots\right) \tag{67}$$

Now we can define the right-hand sides of equations (65–66) to be new variables y, (y_B and y_H) expand the left-hand sides as a series in the variable x defined in Eq. (55) and solve for x in terms of y by inverting the series. Then, we can compare how the two definitions of entropy agree with the exact results obtained from Eq. (46). Using Eq. (50) and Eq. (55) in *Mathematica*, we obtain (here including only the first few terms in y and N^{-1})

$$
\begin{aligned}
x = {} & y\left(1+N^{-1}+\frac{N^{-2}}{3}-\frac{N^{-3}}{3}-\frac{N^{-4}}{45}\right)+y^{3}\left(-\frac{1}{48}-\frac{N^{-1}}{48}+\frac{N^{-2}}{72}-\frac{N^{-3}}{72}-\frac{131N^{-4}}{720}\right)\\
& +y^{5}\left(\frac{1}{1920}+\frac{N^{-1}}{1920}+\frac{N^{-2}}{5760}-\frac{N^{-3}}{5760}-\frac{263N^{-4}}{43200}\right)+y^{7}\left(-\frac{17}{1290240}-\frac{17N^{-1}}{1290240}+\frac{N^{-2}}{967680}-\frac{N^{-3}}{967680}\right)\\
& +y^{9}\left(\frac{31}{92897280}+\frac{31N^{-1}}{92897280}+\frac{N^{-2}}{278691840}-\frac{N^{-3}}{278691840}\right)+O(y^{11})
\end{aligned}
\tag{68}
$$

and

$$
\begin{aligned}
\ln K = {} & y\left(1+N^{-1}+\frac{N^{-2}}{3}-\frac{N^{-3}}{3}-\frac{N^{-4}}{45}\right)+y^{3}\left(\frac{N^{-1}}{24}+\frac{7N^{-2}}{72}+\frac{N^{-3}}{36}-\frac{41N^{-4}}{540}\right)\\
& +y^{5}\left(\frac{N^{-1}}{1920}+\frac{31N^{-2}}{5760}+\frac{59N^{-3}}{5760}-\frac{181N^{-4}}{86400}\right)+y^{7}\left(\frac{N^{-1}}{322560}+\frac{127N^{-2}}{967680}+\frac{173N^{-3}}{241920}\right)\\
& +y^{9}\left(\frac{N^{-1}}{92897280}+\frac{73N^{-2}}{39813120}+\frac{6869N^{-3}}{278691840}\right)+O(y^{11})
\end{aligned}
\tag{69}
$$

These two equations allow us to substitute the two different definitions of y and compare. They are to be compared with exact numerical values where we evaluate Eq. (46) to find N_2, which gives us x exactly using Eq. (55). The variable y in the exact case is

$$
y = -\frac{\Delta E}{kT_e}
\tag{70}
$$

where T_e is the equilibrium temperature and is equal to T_H. Using many series of numerical computations with varying values of N and ΔE and several values of the dimension, we found the first few terms in the series expansion for x in terms of y are given by

$$
\begin{aligned}
x = {} & y\left(1-\frac{2}{dN}\right)+y^{3}\left(-\frac{1}{48}+\frac{(d+1)N^{-1}}{4d^{2}}-\frac{(11d+6)N^{-2}}{12d^{3}}+\frac{N^{-3}}{d^{3}}+O(N^{-4})\right)\\
& +y^{5}\left(\frac{1}{1920}-\frac{(d+1)(d+2)N^{-1}}{64d^{3}}+O(N^{-2})\right)+y^{7}\left(-\frac{17}{1290240}+O(N^{-1})\right)+\cdots
\end{aligned}
\tag{71}
$$

Fits to numerical $\ln K$ results gave

$$
\begin{aligned}
\ln K = {} & y\left(1-\frac{2}{dN}\right)+y^{3}\left(\frac{(d+2)N^{-1}}{8d^{2}}-\frac{(4d+3)N^{-2}}{6d^{3}}+\frac{5N^{-3}}{6d^{3}}+O(N^{-4})\right)\\
& +y^{5}\left(-\frac{(d^{2}+12d+12)N^{-1}}{384d^{3}}+O(N^{-2})\right)+y^{7}\left(O(N^{-1})\right)+\ldots
\end{aligned}
\tag{72}
$$

where some additional terms are known but not included here. Note that all corrections to ln$K$ after the leading y are inversely proportional to N to at least the first power. Note also that many of the coefficients in Eqs. (71–72) are dimension dependent whereas Eqs. (68–69) appear to be dimension independent. Eq. (72) contains the corrections for finite systems we sought.

We seek to compare only the leading correction term in N^{-1}. The leading correction term in the exact results for (ln$K$ - $\Delta E/kT$)N is $-2/dN$ (see Eq. (72)). The leading correction term for the HS entropy is $+1/N$ (see Eq. (68)) and for the BP entropy, it is $(d-2)/dN$ obtained by substituting Eq. (67) into Eq. (69). Thus, neither entropy gives the proper leading correction. The BP entropy is slightly better, but neither one matches the correct value. However, the derivation above did suggest the functional form that allowed the fitting of Eqs. (71–72). In addition, this treatment shows explicitly that both entropies give the correct equilibrium constant in the limit of large N.

The expansions in Eqs. (71–72) converge only for small values of $|\Delta E|$. In particular, when ΔE is large and positive, $\langle N_2 \rangle$ approaches zero. Under these conditions the maximum value of Eq. (42) is at an unphysical value negative value, and the partition function is a decreasing function of N_2. However, when $\Delta E/kT > 1$ and $\langle N_2 \rangle > 3$, the linear prediction without any N^{-1} corrections is a better approximation than the higher polynomial approximation. Later, when $\langle N_2 \rangle$ is les than approximately 3, ln$K$ abruptly changes slope from about 1 to a slope greater than 2. This kind of behavior cannot be approximated by polynomial expansions like those above.

XI. Adiabatic Expansions and Compressions

The science of thermodynamics developed through the macroscopic study of the exchange of heat and work between a sample gas and its surroundings. For example, the Carnot cycle consists of an isothermal expansion of a gas sample, then an adiabatic expansion ending in a lower temperature, followed by an isothermal compression and, finally, an adiabatic compression ending at the initial temperature. Isothermal changes of ideal gas samples are easily understood. However, adiabatic changes are more difficult.

The differential equation describing adiabatic changes using macroscopic parameters for an ideal gas is

$$dE = nc_v dT = dq + dw = 0 - P_{ext} dV = -\frac{nRT}{V} dV \qquad (73)$$

where we have equated the total differential of energy to the differential work with the assumption of no heat transfer for an adiabatic change. Temperature and volume dependence in Eq. (73) can separated and integrated to obtain

$$\frac{T_2}{T_1} = \left(\frac{V_1}{V_2} \right)^{R/C_v} \qquad (74)$$

For a monatomic ideal gas in three dimensions, $c_v = 3R/2$, and the exponent of the volume ratio is 2/3. This adiabatic equation of state relates the initial and final temperatures to the initial and final volumes. The assumptions made are: (1) ideal gas behavior, (2) no heat transfer, and (3)

$P_{ext} = P$ for reversibility. Assumption 2 is obeyed better for a fast process, and assumption 3 is obeyed better for a slow process, but there is often a range of motion for which both assumptions are good.

Boltzmann 3D uses accurate hard-sphere dynamics to demonstrate many of the principles of kinetic theory in the classroom and the teaching laboratory. The simulation of adiabatic expansions and compressions is one of its useful capabilities. This section answers the following question: How can the temperature change in an adiabatic change be understood in terms of the microscopic kinetic theory of gas molecules?

Consider a piston of mass M serving as the top of a container and confining a gas of particles each of mass m. The piston collides with individual gas molecules as it moves in the z (vertical) direction with velocity V_1. Each collision will conserve the z-component of linear momentum and (twice) the z-component of kinetic energy of the system composed of the piston and one colliding particle:

$$MV_1 + mv_{1z} = MV_2 + mv_{2z} \tag{75}$$

$$MV_1^2 + mv_{1z}^2 = MV_2^2 + mv_{2z}^2 \tag{76}$$

where subscript 1 refer to velocities before collision, and subscript 2 refers to after the collision. Solving Eq. (75) for V_2, substituting into Eq. (76), and solving the resulting quadratic equation for v_{2z}, we obtain

$$v_{2z} = \frac{2V_1 - v_{1z}\left(1-\frac{m}{M}\right)}{1+\frac{m}{M}} \xrightarrow[M>>m]{} 2V_1 - v_{1z} + O\left(\frac{m}{M}\right) \tag{77}$$

In choosing the physical root of the quadratic, we have used $v_{1z} > V_1$, otherwise no collision will occur. Using Eq. (77) in Eq. (75),

$$V_2 = \frac{V_1\left(1-\frac{m}{M}\right) + 2v_{1z}\frac{m}{M}}{1+\frac{m}{M}} \xrightarrow[M>>m]{} V_1 + O\left(\frac{m}{M}\right) \tag{78}$$

If we assume the piston is much heavier than the individual particles, we can neglect terms of order m/M and higher. Under these conditions, Eq. (78) states that the motion of the piston is unchanged by the collision. However, the neglected terms describe kinetic energy and momentum transfer to the piston from the particle due to the collision. Accumulation of the effects of a large number of such collisions will require a force to keep the piston moving at a constant speed, and thus work will be exchanged with the gas in the piston. On the other hand, momentum and energy transfer from the piston to the colliding gas molecule is explicit in Eq. (77). If V_1 is negative (compression), the kinetic energy and the magnitude of the linear momentum of a particle after the collision is larger than it would be in an elastic collision. If V_1

is positive (expansion), the particle has decreased kinetic energy and magnitude of linear momentum. One can now show that, when averaged over the distribution of molecular velocities, the average change in kinetic energy is consistent with the macroscopic equations, (73) and (74).

The change in kinetic energy for a particle of initial velocity *z*-component, v_{1z} is, using Eq. (77);

$$\tfrac{1}{2}m(2V_1 - v_{1z})^2 - \tfrac{1}{2}mv_{1z}^2 = -2mV_1(v_{1z} - V_1) \tag{79}$$

We can now average this change over the distribution of initial collision velocities in a gas, $P(v_{1z})$. The change in the gas kinetic energy, *dK*, in time interval *dt* is given by

$$dK = dt\,A_d \frac{N}{V_d}\int_{V_1}^{\infty}\left[-2mV_1)(v_{1z} - V_1)\right](v_{1z} - V_1)P(v_{1z})dv_{1z} \tag{80}$$

where A_d is the *d*-dimensional area of the piston (area for $d = 3$, length for $d = 2$, and unity for $d = 1$), V_d is the *d*-dimensional volume over which *N* particles are distributed, and $(v_{1z} - V_1)$ is the rate at which particles approach the piston, so $(v_{1z} - V_1)dt$ is the distance over which particles travel relative to the moving piston during time *dt*. The lower limit of the integral is V_1 because particles are moving away from the piston if $v_{1z} < V_1$. Equation (80) assumes that the particle concentration, N/V_d, is unchanged by the motion of the piston. This will only be true if V_1 is small compared to typical particle speeds. For example, if V_1 is large and positive, particles do not have time to move into the region vacated by the piston. On the other hand, if V_1 is large and negative, the distribution of molecular velocities does not have time to relax to an equilibrium distribution, and in addition, particles will stick to and be pushed by the piston. We will assume that the number of particles is large so $P(v_{1z})$ can be taken to be the Maxwell-Boltzmann distribution for one velocity component. The integral in Eq. (80) cannot be evaluated exactly with an arbitrary lower limit, so we will separate the integral into two parts.

$$\begin{aligned} dK &= -2mV_1 dt\frac{N}{L_z}\int_{V_1}^{\infty}(v_{1z} - V_1)^2 P(v_{1z})dv_{1z} \\ &= -2mV_1 dt\frac{N}{L_z}\left[\int_{V_1}^{0}(v_{1z}^2 - 2v_{1z}V_1 + V_1^2)P(v_{1z})dv_{1z} + \int_{0}^{\infty}(v_{1z}^2 - 2v_{1z}V_1 + V_1^2)P(v_{1z})dv_{1z}\right] \end{aligned} \tag{81}$$

where we have simplified using $V_d = A_d L_z$, where L_z is the instantaneous height of the piston above the container floor. In writing Eq. (81), we have also assumed V_1 is negative (compression), but the resulting formula will also apply for positive values of V_1 (expansion) because the sign of the integral reverses when the limits are reversed. Since V_1 is small, we can approximate the first integral by approximating $P(v_{1z})$ within the integral by $P(0)$, the accuracy of which will be verified later. The second integral can be evaluated exactly. Using $P(v_{1z}) = [m/(2\pi kT)]^{1/2}\exp[-mv_{1z}^2/(2kT)]$, we obtain:

$$dK = \frac{-2mV_1 N dt}{L_z}\left(\frac{m}{2\pi kT}\right)^{1/2}\left[-\frac{V_1^3}{3} + \left(\frac{\pi}{2}\right)^{1/2}\left(\frac{kT}{m}\right)^{3/2} - 2V_1\frac{kT}{m} + V_1^2\left(\frac{\pi}{2}\right)^{1/2}\left(\frac{kT}{m}\right)^{1/2}\right] \tag{82}$$

In this formula, we define a typical velocity, $v_{typ} = (kT/m)^{1/2}$, proportional to the rms and average velocity (the proportionality constants vary for $d = 3, 2, 1$), and arrange the terms in increasing powers of V_1/v_{typ}

$$dK = -\frac{V_1 kTN\,dt}{L_z}\left[1 - 2\left(\frac{2}{\pi}\right)^{1/2}\frac{V_1}{v_{typ}} + \pi^{1/2}\frac{V_1^2}{v_{typ}^2} - \frac{1}{3}\left(\frac{2}{\pi}\right)^{1/2}\frac{V_1^3}{v_{typ}^3}\right] \tag{83}$$

Here, the first term comes from the v_{1z}^2 average, and the last term comes from the integral from V_1 to 0. We now make the approximation that V_1 is much smaller than v_{typ} and retain only the leading term in Eq. (83). The fact that these higher terms are negligible justifies the approximation made in evaluating the integral above. Requiring $V_1 << v_{typ}$ also avoids the kinds of problems discussed earlier, since during time dt, particles have time to occupy the volume vacated by the piston, and the particles average many collisions in the time the piston moves the distance of one particle mean free path ($\lambda = v_{avg}\tau$).

In Eq. (83), $K = dNkT/2$ for a monatomic gas, so $dK = (d/2)NkdT$ and $c_V = dNk/2$. Making the substitution $d/2 = c_V/R$, along with $V_1dt = dL_z$, for a mole of gas, we arrive at the differential equation

$$\frac{c_V}{R}\frac{dT}{T} = -\frac{dL_z}{L_z} = -\frac{dV}{V} \tag{84}$$

which is equivalent to Eq. (73) and has solution Eq. (74).

Thus, the collective effect of kinetic energy transfer by a moving piston on the individual gas molecules due to their collisions with piston is, in the limit of pistons moving more slowly than typical molecular speeds, precisely what is expected by the macroscopic laws of thermodynamics. This is guaranteed by the principle of equipartition of energy in that the average kinetic energy in the z direction is $kT/2$ per particle. And since the kinetic energy is an even function of v_z, the average kinetic energy of those particles traveling in the direction opposite of the piston, is $kT/4$ per particle.

It is useful to note that the velocity distribution for a gas with a finite number of particles can differ somewhat from the Maxwell-Boltzmann distribution; however, the result above is still exact for any finite system. This is because the leading term involves the average of v_{1z}^2, which has the same value for any finite system due to the equipartition principle for kinetic energy. However, there may be a small correction in the coefficient of the second term of Eq. (83) for a finite system.

We have performed simulations of molecular motion for a monatomic gas using the computer program Boltzmann 3D in 1, 2 and 3 dimensions, confirming the accuracy of this result in the limit of piston velocity much slower than typical molecular velocity. We are unable to verify the accuracy of the correction terms given by Eq. (83) or the difference between the values for different simulation dimensions.

It is interesting to note that the change in kinetic energy is independent of the precise value of V_1 as a function of time as long as it is always small compared to a typical particle speed. In other words, only the final and initial positions are required, not the behavior of $L_z(t)$. This result is related to the fact and energy and entropy are state functions. Microscopically, the amount of energy transferred is a function only of the piston logarithmic ratio dL_z/L_z.

It has been long understood from the law of conservation of energy that compression of a gas increases the kinetic energy and gas temperature because of the work done on the system and that expansion of a gas decreases its kinetic energy and temperature because of the work done by the gas on the surroundings. This derivation shows how this result can be quantitatively understood in terms of kinetic theory of molecular motion.

When one performs an adiabatic compression in **Boltzmann 3D** followed by an expansion to the original volume, Eq. (74) predicts that the final temperature is the same as the initial temperature. You will find that the final temperature will not necessarily be the same. The reason for this is that, for a finite system moving at finite speeds, the distribution of collision velocities does not accurately represent the entire distribution. As the number of particles is increased and the rate of piston motion is decreased, the range of final temperatures comparing repeated identical experiments and the difference between the initial and final temperatures will decrease. On the average (it is possible for exceptions to occur), the final temperature will be higher than the initial temperature after an adiabatic cycle. One can interpret this as a result of the Second Law of Thermodynamics. The final entropy of the system of hard spheres has increased due to the higher final temperature, while the surroundings have been returned to their original state, so the entropy of the universe has increased.

XII. Description of Sample Files

This section describes each of the sample demonstration files as they are supplied in the download package. These files can be used for demonstrations, or they can be altered by the user. In particular, you will want to adjust the arena size and the zoom factor in each file so that each demonstration will fit properly in your display device at the screen resolution you prefer.

A. default.set: 50 2D particles of mass 2 amu, radius 1 nm, 300 K. Simulates room temperature H_2 gas.

B. 3Dhyd100.set, 2Dhyd100.set: 50 particles of mass 2 amu (can be reset to 2.016 if desired), radius 0.7 nm, 100 K. Gives velocity distribution of H_2 at low temperature.

C. 3Dhyd1000.set, 2Dhyd1000.set: 50 particles of mass 2 amu (can be reset to 2.016 if desired), radius 0.7 nm, 1000 K. Gives velocity distribution of H_2 at high temperature.

D. 3Doxy100.set, 2Doxy100.set: 50 particles of mass 32 amu, radius 1.1 nm, 100 K. Gives velocity distribution of O_2 at low temperature.

E. 3Doxy1000.set, 2Doxy1000.set: 50 particles of mass 32 amu, radius 1.1 nm, 1000 K. Gives velocity distribution of O_2 at high temperature.

F. 3Dmixture.set, 2Dmixture.set: 30 red particles of mass 2, radius 0.7 nm and 30 blue particles of mass 32, radius 1.1 nm. Shows difference between velocity distribution of heavy and light particles even though they have identical energy distributions.

G. 3Ddiffuse.set, 2Ddiffuse.set: 100 red particles of mass 2, radius 2 nm and 1 blue particle of identical mass and radius. For observing diffusion of the blue particle among the red ones. Uses periodic boundary conditions.

H. 3Deffuse.set, 2Deffuse.set 30 red particles of mass 2, radius 0.7 nm, and 30 blue particles of mass 32, radius 1.1 nm, all on the left of a divided arena with an opening. It loads in a paused state. Click the **resume** button to see if Graham's Law is obeyed with the red ones effusing four times faster than the slower blue ones.

I. 3Dhotcold.set, 2Dhotcold.set: 30 red particles of mass 2, radius 0.7 nm, temperature 1200 K on the left of a divided arena and 30 blue particles of identical mass and radius but at temperature 300 K. Examine the difference between their velocity and energy distributions. Show how pressure is proportional to temperature. Open a hole in the barrier or remove it altogether to see energy and temperature come to equilibrium.

J. 3Dbrownian.set, 2Dbrownian.set. One red particle of mass 200 amu, radius 5 nm with 200 blue particles with mass 1, radius 0.2 nm. Watch the stochastic motion of the large red particle due to impacts from the small blue particles. Uses periodic boundary conditions.

K. 3Dentropy.set, 2Dentropy.set: 24 motionless blue particles of mass 2 amu and radius 1.4 nm arranged at the lower left corner of the arena with one identical blue particle at the upper right with high kinetic energy (T = 9999K). It loads in a paused state. Click the **resume** button to see the moving particle collide with others and distribute its energy eventually to all of the others. They also move to random positions within the arena.

L. 3Dpresnum.set, 2Dpresnum.set: 40 red particles in the left half and 20 blue particles in the right half, all with mass 2 amu and radius 1 nm, showing pressure is proportional to number of particles.

M. 2Dequal.set. 20 particles on the left and 20 particles on the right of a 2D divided arena, each with mass 2 amu and radius 1 nm. Each particle on the right has the identical velocity and corresponding position with a particle on the left. Watch the particles follow the same motions until chaotic dynamics causes round-off error differences between the two systems to exponentially diverge until visible differences emerge after about 95 ps and they are completely different by 120 ps.

N. 2Dliquid.set. 47 particles with radius 6 nm and mass 2 amu and one blue particle in periodic boundary conditions and trajectory trace mode. Press start to see the particles trace out their trajectories. For estimating diffusion constants in a dense fluid.

O. 3Dequil.set, 2Dequil.set. 50 red particles with radius 1 nm and mass 2 amu in reaction mode with default values of activation energy. The time-averaged, equilibrium number of blue particles is predicted to be 13.3 in 2D and 14.1 in 3D.

XIII. A Poem by Roald Hoffman Describing the Maxwell-Boltzmann Distribution

Heat:Hot, as _____:Cold

A poem by Roald Hoffman (used by permission), published in Roald Hoffmann and Vivian Torrance, *Chemistry Imagined, Reflections on Science* (Smithsonian Institution Press, Washington, D.C., 1993), 46.

Deep in,
they’re there, they’re
at it all the time, it’s jai
alai on the hot molecular fronton-
a bounce off walls onto the packed aleatory
dance floor where sideswipes are medium of exchange,
momentum trades sealed in swift carom sequences,
or just that quick kick in the rear, the haphaz-
ard locomotion of the warm, warm world.
But spring nights grow cold in Ithaca;
the containing walls, glass or metal,
are a jagged rough rut of tethered
masses, still vibrant, but now
retarding, in each collision,
the cooling molecules.
There, they’re there,
still there,
in deep,
slow
.

References

B. J. Alder and T. E. Wainwright, *J. Chem. Phys.* **27**, 1208–1209 (1957).

B. J. Alder and T. E. Wainwright, *J. Chem. Phys.* **31**, 459–466 (1959).

B. J. Alder and T. E. Wainwright, *J. Chem. Phys.* **33**, 1439–1451 (1960).

B. J. Alder, *J. Chem. Phys*. **40**, 2724–2730 (1964).

B. J. Alder and T. Einwohner, *J. Chem. Phys.* **43**, 3399–3400 (1965).

B. J. Alder, W. G. Hoover, and D. A. Young, *J. Chem. Phys*. **49**, 3689–3697 (1968).

B. J. Alder and C. E. Hecht, *J. Chem. Phys*. **50**, 2032–2037 (1969).

T. Einwohner and B. J. Alder, *J. Chem. Phys*. **49**, 1458–1473 (1968).

N. Clisby and B. M. McCoy, preprint, Condensed Matter Physics archives #0303101 (20 Mar 2003).

F. Lado, *J. Chem. Phys.* **75**, 5461 (1981)

I. E. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products* (Academic Press, New York, 1965).

Handbook of Chemistry and Physics, ed. D. R. Lide (Chemical Rubber Company, Boca Raton, published annually).

W. G. Hoover and B. J. Alder, *J. Chem. Phys.* **46**, 686–691 (1967).

D. A. McQuarrie, *Statistical Mechanics* (Harper and Row, New York, 1973).

F. L. Roman, A. Gonzalez, J. A. White, S. Velasco, *Physica A* **234**, 53–75 (1996).

R. B. Shirts, S. R. Burt, and A. M. Johnson, *J. Chem. Phys*. 125, 164102 (2006).

W. W. Wood and J. D. Jacobson, *J. Chem. Phys*. **27**, 1207–1208 (1957).