

Applet 1

First, as the results of these simulations are generally time-averaged values and there is no provided benchmark timeframe as to what should constitute a sufficient timeframe over which to simulate these various conditions, several trials were done to see how the results of one condition change from incrementing only the simulation time. The simulation was run for $N=50$, isothermal temperature $=1.0$, and timeframes of 1E3, 2E3, and 3E3. The results of these trials can be seen as summarized in **FIG. 1**.

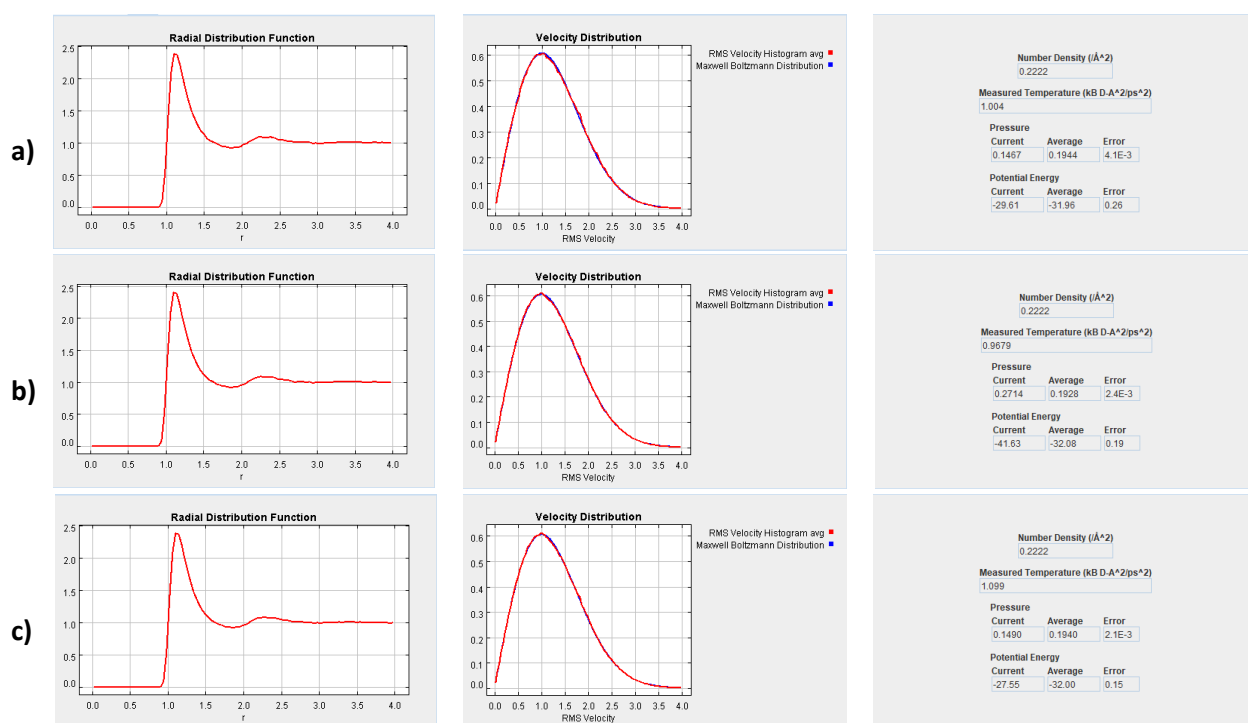


FIG. 1. The Radial Distribution Function, Velocity Distribution, and various Metrics are displayed for the results of simulation times of **a) 1E3 b) 2E3 and c) 3E3**.

While the Radial Distribution Function $g(r)$ can appear similar between the time trials, the Velocity Distribution and Metrics provide more insight into the accuracy of the time-averaged results. The Velocity Distribution plot for a timeframe of 1E3 was noted to have not yet sufficiently converged to the expected Maxwell Boltzmann Distribution. The Metrics of the same trial gave errors of 4.1E-3 for pressure and 0.26 for potential energy. The Velocity Distribution plot for a timeframe of 2E3 was seen to have better convergence onto the expected Maxwell Boltzmann Distribution. The Metrics of this second same trial resulted in errors of 2.4E-3 for pressure and 0.19 for potential energy. And the Velocity Distribution plot for a 3E3 timeframe was displayed to be similarly converged as the 2E3 case, with errors of 2.1E-3 and 0.15 for pressure and potential energy, respectively. The results of these preliminary trials were taken to support a timeframe of 2E3 as being sufficient, it is assumed in this report that 2E3 will be a sufficient simulation time for all simulations conducted – that is, that all time-averaged properties will provide valid results within this timeframe.

a) Isothermal Density Study

For this study, an isothermal case is considered with a temperature input value of 1.0, over varied densities. The impact this varied density has on the Radial Distribution Function is then considered. Values of N were varied as $N=50$, $N=100$, $N=150$, and $N=200$ over a timeframe of $2E3$, as all subsequent simulations will be. Incrementing N over the given area of the simulation will provide a direct incrementation of density. The $g(r)$ plots for these conditions can be seen in **FIG. 2**.

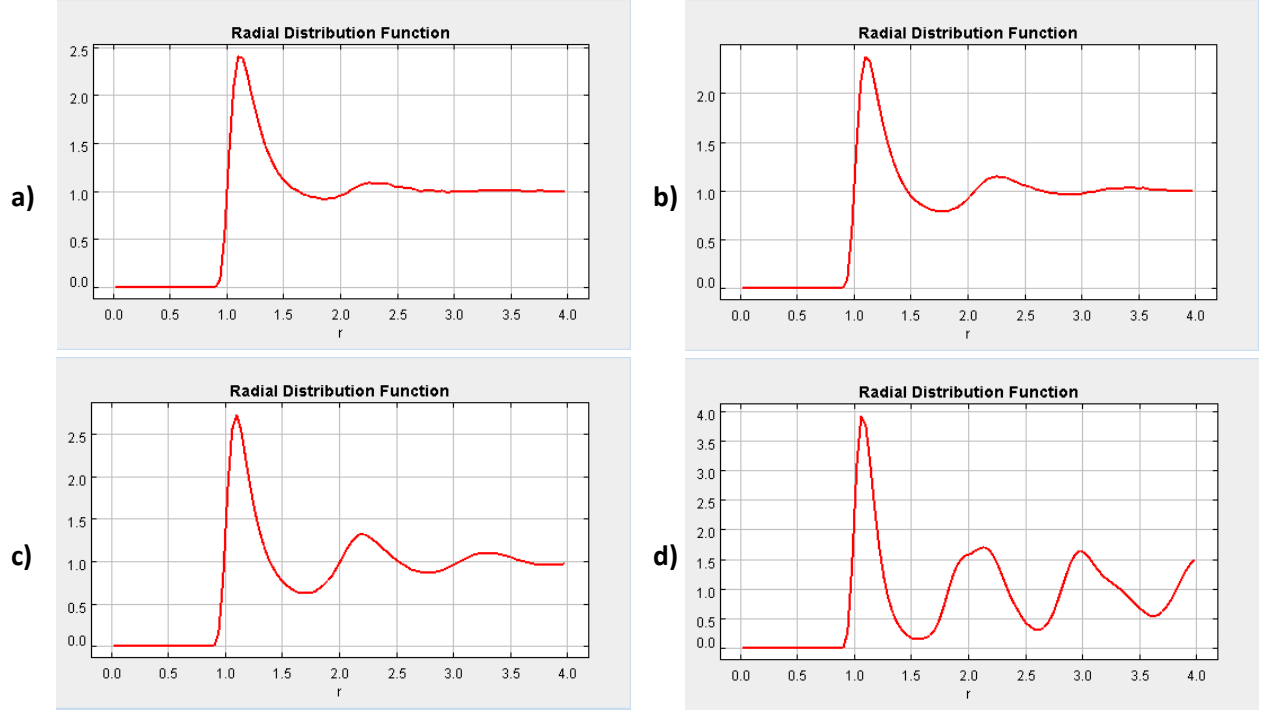


FIG. 2. The Radial Distribution Function is plotted against increasing radius for various densities of **a)** $N=50$ **b)** $N=100$ **c)** $N=150$ **d)** $N=200$.

These results display $g(r)$ as having an increasing radial variance from 1 as the value of N is positively incremented. With increasing density, the Radial Distribution Function has larger amplitude of variance, and requires a significantly greater radial distance to approximately converge to a value of 1. This makes physical sense and matches expectations of this hard sphere fluid study, as more dense configurations are more confined at a given energy. For the cases of higher densities, the particles have less relative movement and the gaps in between particles are more static. This leads to a more sporadic radial distribution with undulations of greater magnitude.

$$\rho(r) = \langle \rho \rangle g(r) \quad (1)$$

$$N \equiv \langle \rho \rangle \int 4\pi r^2 g(r) dr = \langle \rho \rangle V \quad (2)$$

Local density is given in equation (1), and it is apparent that as the radial density approaches the average density of the fluid, $g(r)$ approaches 1. Under the hard sphere assumptions made in this simulation, among them being that $g(r)$ is only a function of r and not impacted by particle orientations (that the impact of orientation averages out) and that the particles are indistinguishable, it is apparent that equation (2) supports these results over the given simulation *area*. Summarized, at low densities the time-averaged spatial density of the fluid, as simulated, is more radially regular (and closer to the average) than a high density fluid – and with fewer interactions between particles. The simulation of these aspects of low density fluids lead to $g(r)$ converging to 1 over a shorter radial distance. The configurations sampled by this ensemble are described by the Grand Canonical Ensemble.

b) Isothermal Temperature Study

This study considers how isothermal temperature impacts the Radial Distribution Function, as simulated. Here, simulations are run over varied temperature for a set density, then density is incremented and temperatures are again varied. The same densities that were tested in part a) are again chosen for density increments. Temperature is varied over values of 0.5, 1.0, 1.5, and 5.0. The results of this study are summarized below in FIG. 3, for the case of $N=100$. For more information on how temperature impacts these simulations, see the Supplemental Information [1].

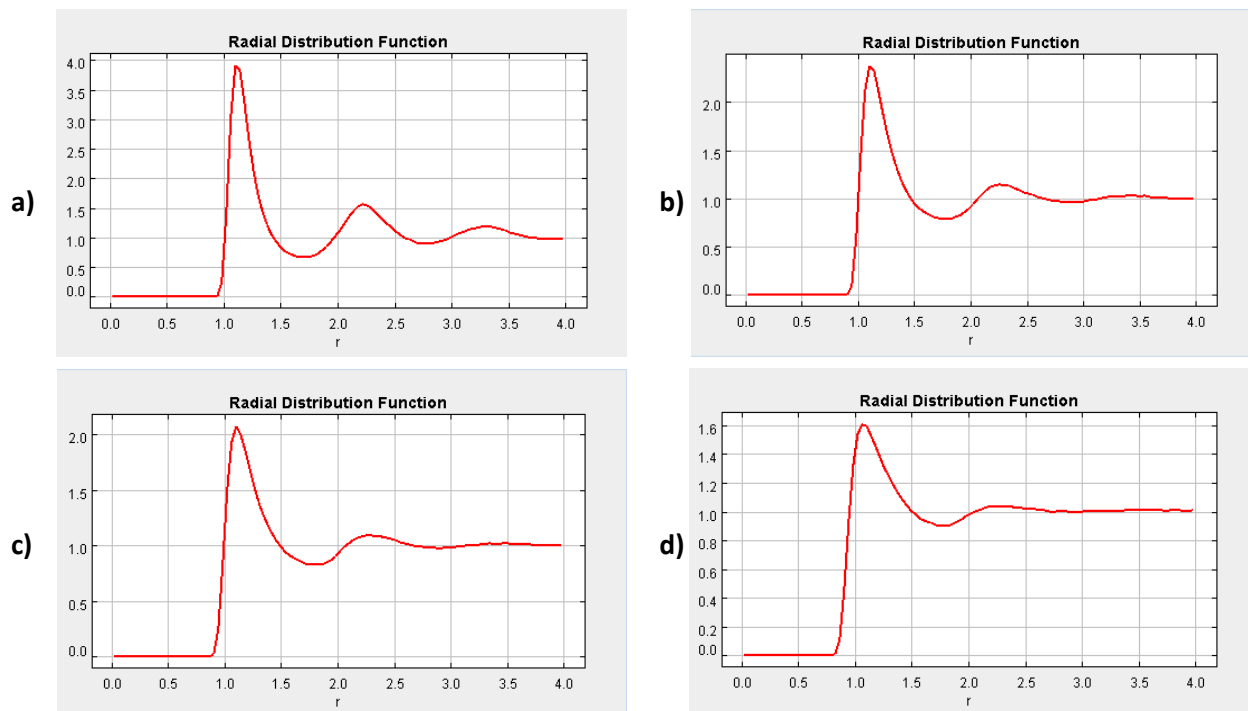


FIG. 3. The results of the simulations of Radial Distribution Function for $N=100$ are plotted as a function of radius and varied by temperatures of a) 0.5 b) 1.0 c) 1.5 d) 5.0.

The results of this isothermal temperature study display that the Radial Distribution Function converges to 1 over a shorter radial distance for increased temperature. The higher temperatures that were simulated see smaller magnitude undulations in their Radial Distribution Function at a given radius. As temperature is increased, particle velocity within the system has a higher average value and a wider probability distribution [2]. With increasing temperature, it is also noted that fluctuations in total energy of the system become more irregular, driven by a more time-disperse potential energy and a more time-sporadic kinetic energy [3].

The results, as summarized above in FIG. 3, support the expectation that at higher temperatures, $g(r)$ will approach a steady value over a shorter radial distance. A higher temperature will create a system with a higher average particle root mean square velocity (or vice versa for a real system). The particles in the system will also have a wider probability distribution of their velocity. An elevated temperature will, in effect, create more “random” movements of the system’s particles, and do so more often. As such, the system’s local density will more approximately approach the system’s average density, and the Radial Distribution Function will converge to 1 over a shorter radial distance.

c) Adiabatic Temperature Study

This study considers how temperature impacts the Radial Distribution Function, for an adiabatically enclosed system. To investigate this, simulations were run over varied temperatures for set densities, then density was incremented and temperatures were again varied. The same densities that were tested above in part a) and b) are again chosen to be density increments. The adiabatic case is selected and energy is varied over values of 0.5, 1.0, 1.5, and 5.0. The subsequent results are summarized in FIG. 4 below, for the case of N=100.

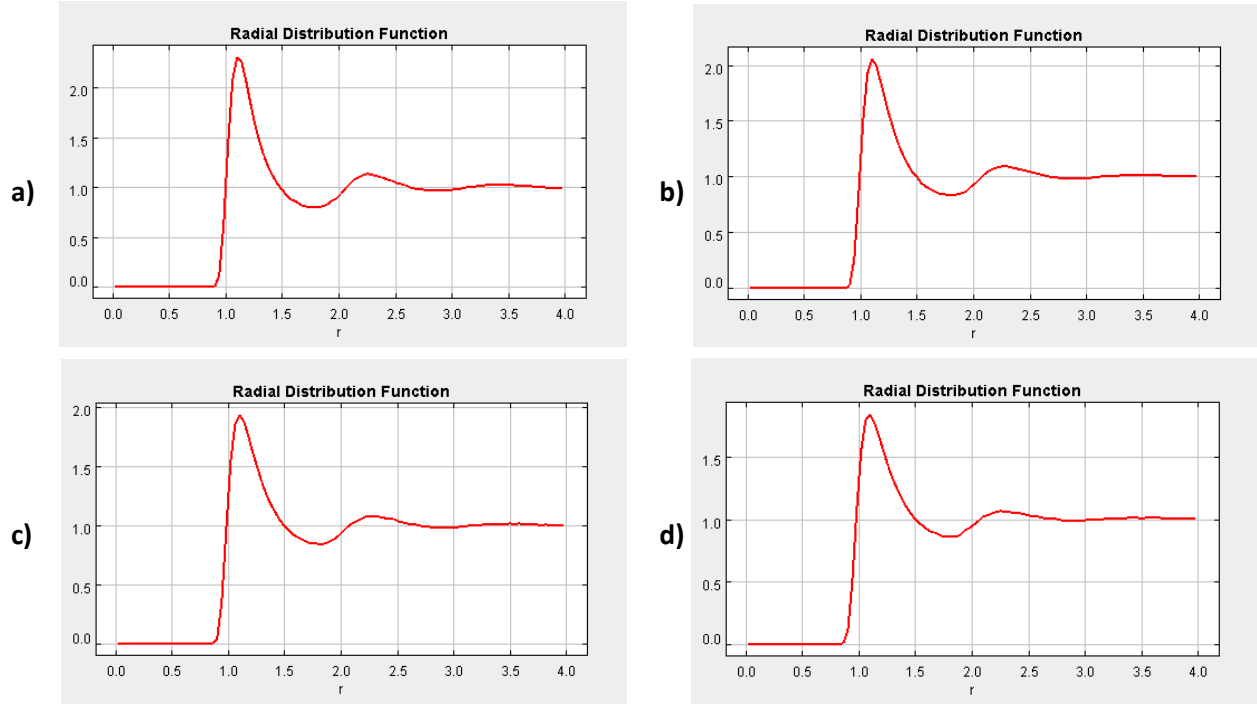


FIG. 4. The results of the simulations of Radial Distribution Function for the adiabatic case of N=100 are plotted as a function of radius and varied by energies of a) 0.5 b) 1.0 c) 1.5 d) 5.0.

In FIG.4 above, it is evident that the Radial Distribution Function's undulations are marginally dampened for the adiabatic case by increasing the temperature. The difference between the isothermal case in part b) and the adiabatic case is most significant at low temperatures. The adiabatic case, as explored here in part c) generally sees the same differences from part a) as part b). It is anticipated that the adiabatic case would see similar results to those presented in part a) if the particle density were varied. The set of configurations sampled in these simulations is described by the Microcanonical Ensemble.

$$\frac{K}{N} = \frac{dkT}{2} \quad (3)$$

$$T = \frac{2K}{dkN} \quad (4)$$

The equipartition theorem in equation (3) provides a way to calculate the temperature over several sampling timeframes during the simulation. First, as to eliminate the likelihood of picking an extraneous value of kinetic energy, four averages were taken over four individual segments of roughly 0.05E3 throughout the total simulation time of 2E3. This was done to see if there is a time dependence in the segment-averaged value of kinetic energy, not just the relatively small variation in kinetic energy within those time segments – that is, if there is an overall trend apart from the fluctuations. These

values of kinetic energy were later input to the equipartition theorem, with N=100 and d=2 for the two-dimensional simulation, to calculate values to temperature that correspond to the sampled kinetic energies [4]. Over the various energies considered (0.5, 1.0, 2.5, 5.0), there was no overall trend identified; however, the kinetic energy of the system does vary with time. The total energy of the system remains constant, but the system does experience relatively small exchanges between potential energy and kinetic energy [5]. This matches expectations of an adiabatically enclosed, rigid system, and is attributed to various interactions between the particles within the system. In FIG. 5 below, the temperatures calculated using equation (4) are shown for energy input of 1.0 and N=100.

Pause time	Avg Kinetic	T (K)
5.00E+02	193.1664	22409.1
1.00E+03	193.5839	22457.53
1.50E+03	193.7905	22481.49
2.00E+03	194.3276	22543.81

FIG. 5. Temperature is calculated using the equipartition theorem at various sampled times.

The values calculated for T are quite consistent; the Percent Relative Standard Deviation (%RSD) is found to be 0.249%. From this, and the above results, it is determined that Temperature can be taken to be constant, excluding of interaction potential-kinetic energy exchange. The bulk temperature of the system is constant, even though the local temperature is

It is noted that the above values of T are quite high, but error in their calculation cannot be identified. Assuming a factor in equation (4) was used incorrectly, the results detailed here are still sound conclusions – with T scaling and %RSD remaining the same. The units are worked out in equation (5) below. It is given that $K [=] [k][T]$ (Jordan said so when asked).

$$T = \frac{2K}{dkN} \quad (4)$$

$$T [=] \frac{\left(\frac{eV}{K} * K\right)}{\left(\frac{eV}{K}\right)} [=] K \quad (5)$$

Using $k = 1.38E - 23 \frac{kg \ m^2}{K \ s^2}$ would make things worse. Finding the error source and calculating realistic values of T is left as an exercise for the reader.

Applet 2

a) Total Steps and Frequency

First, the “Total Monte Carlo Steps” parameter was varied, with all others being held constant, to investigate its impact on the simulation. With a frequency of 1, the total steps were set to values of 50, 100, 200, and 500. The results of each step iteration’s simulation are provided below in FIG. 6.

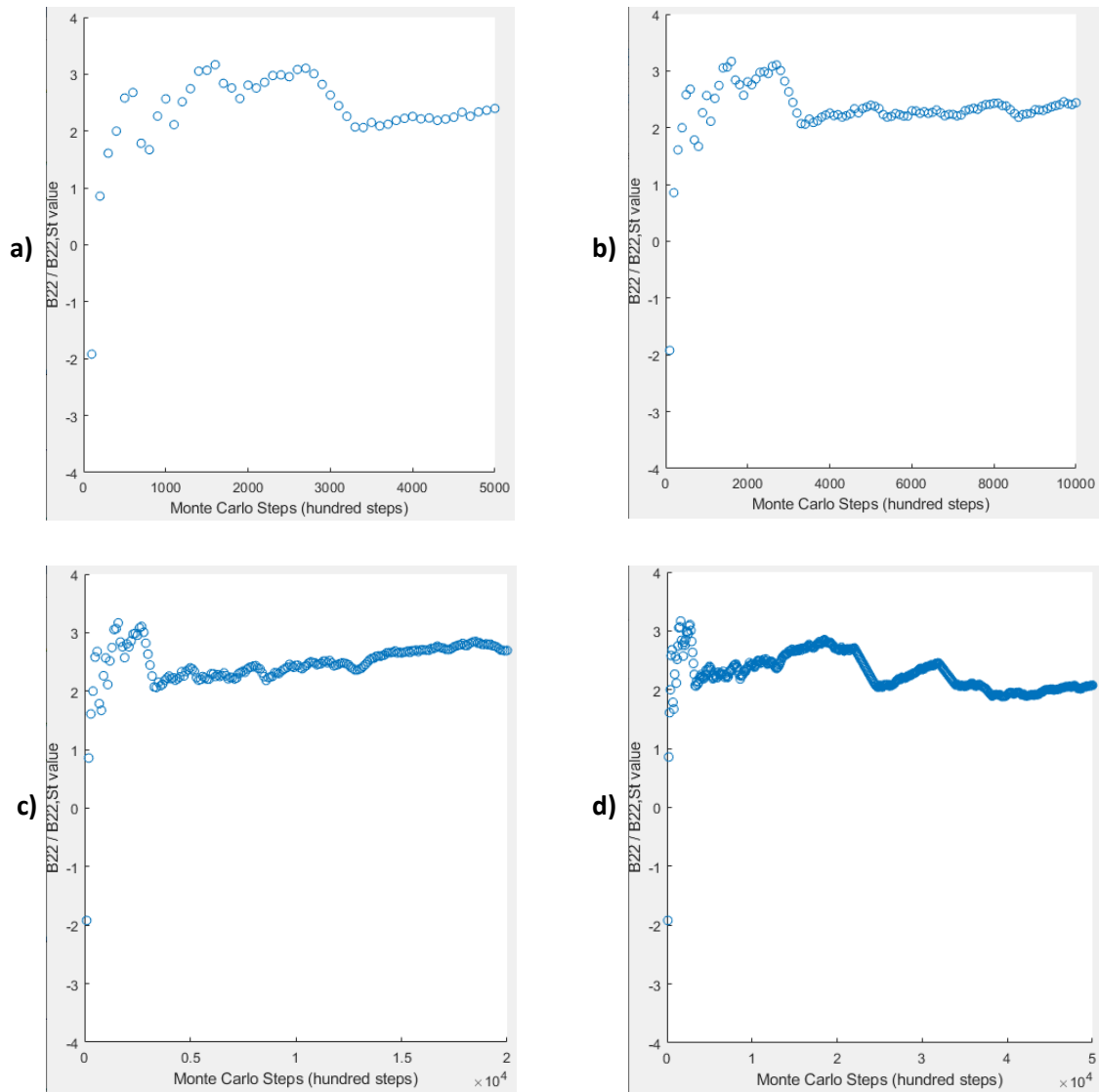


FIG. 6. The Monte Carlo simulation is run for a frequency of 1 and total steps varied as **a)** 50 **b)** 100 **c)** 200 **d)** 500.

From the above results, it is apparent that the Total Monte Carlo Steps parameter controls the total amount of sequential steps taken in the Monte Carlo simulation. In effect, it represents the total number of configurations the simulation will iterate through and produce a data point toward the result. This does match what I would intuitively understand the parameter input to represent, as the “steps” in Monte Carlo simulations are the sets of calculations attempted until accepted by the given criteria. Each set of accepted calculations would produce a data point toward the result, and in this simulation is designated a “step.” It is the total number of data points generated at the given frequency.

Next, the “Frequency” parameter was varied, with all others being held constant, to investigate its impact on the simulation. With a Total Steps value of 100, the frequency was varied as 0.5, 1, 2, and 5. The results of each frequency iteration’s simulation are provided below in FIG. 7.

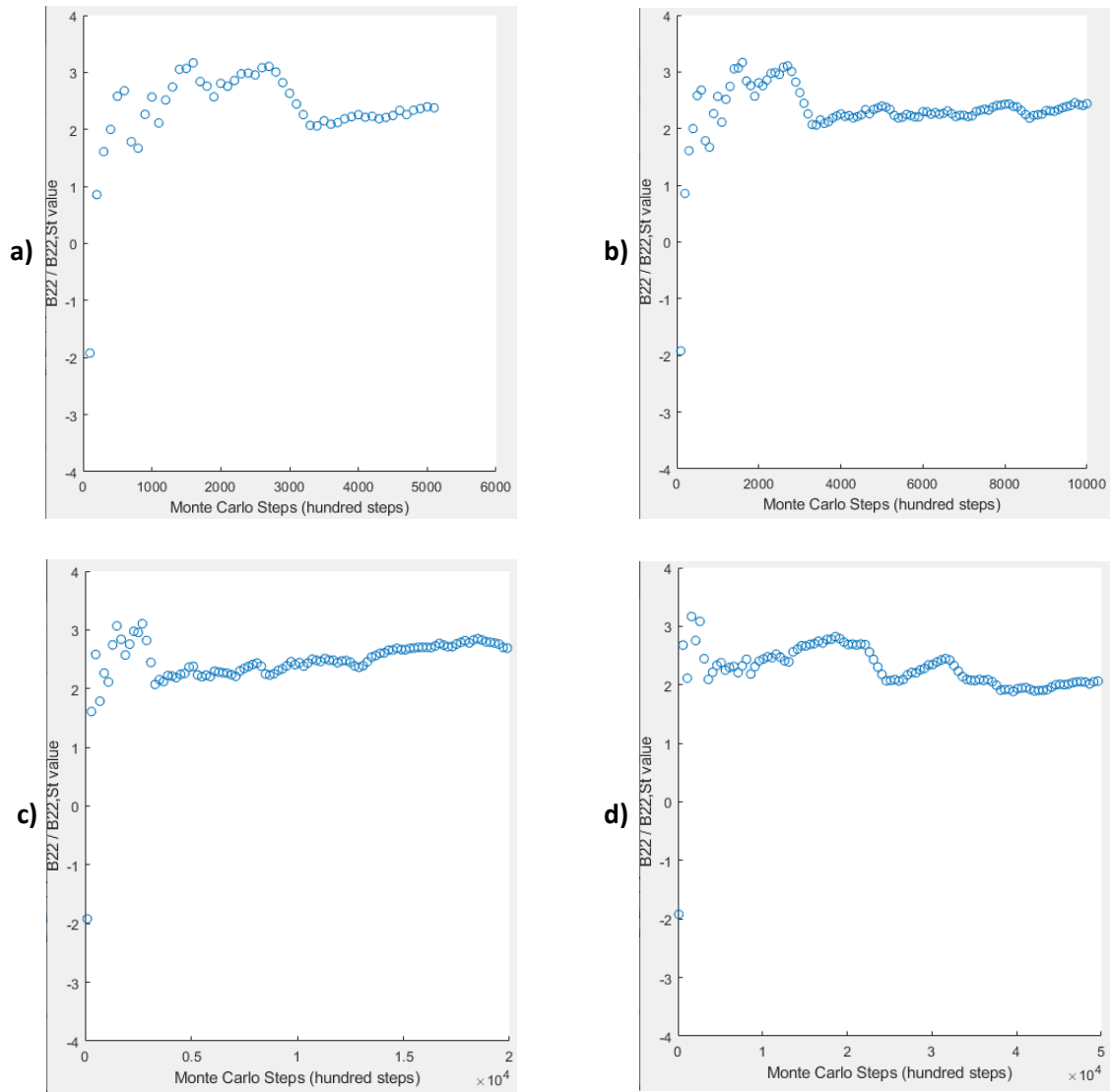
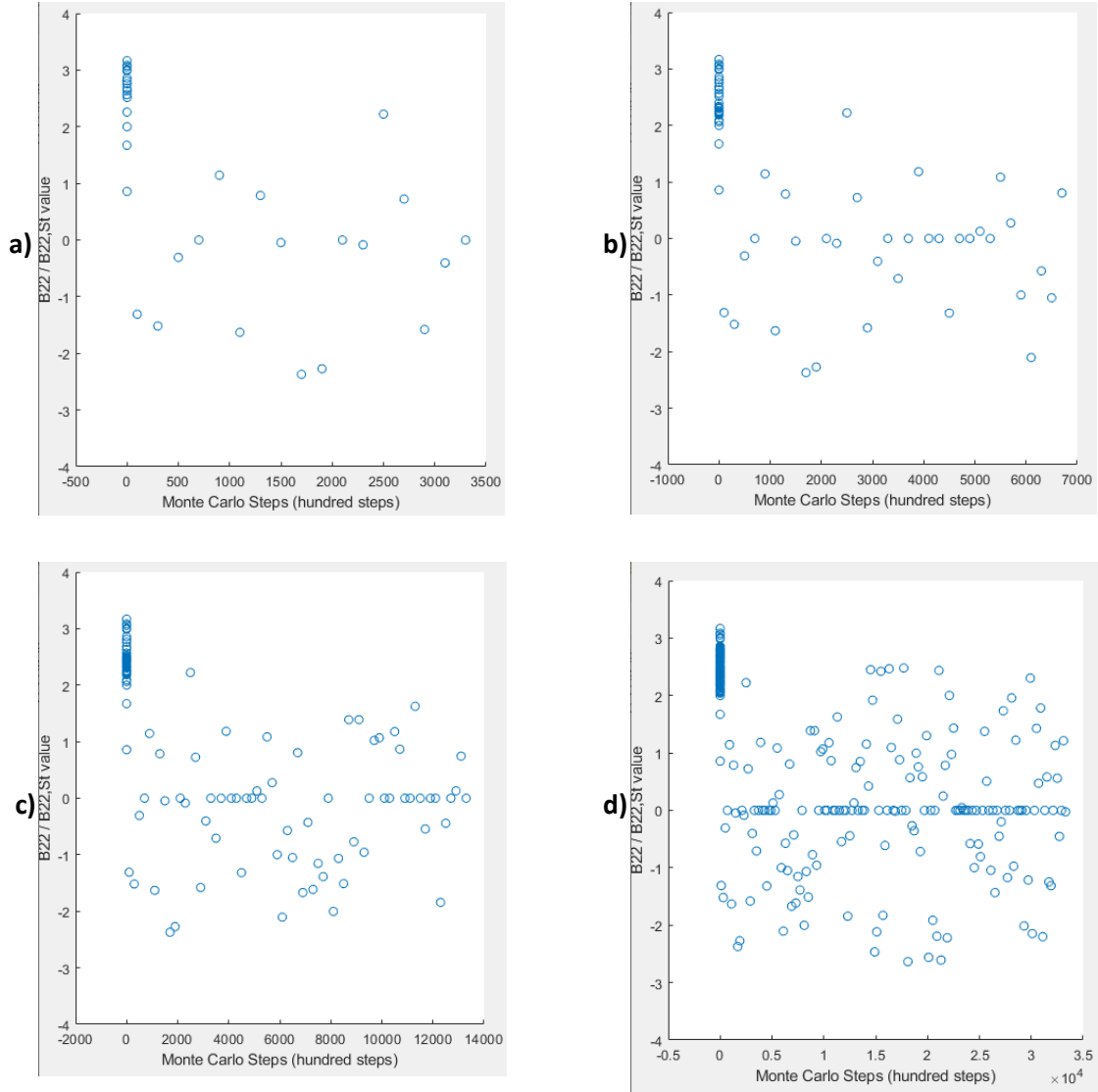


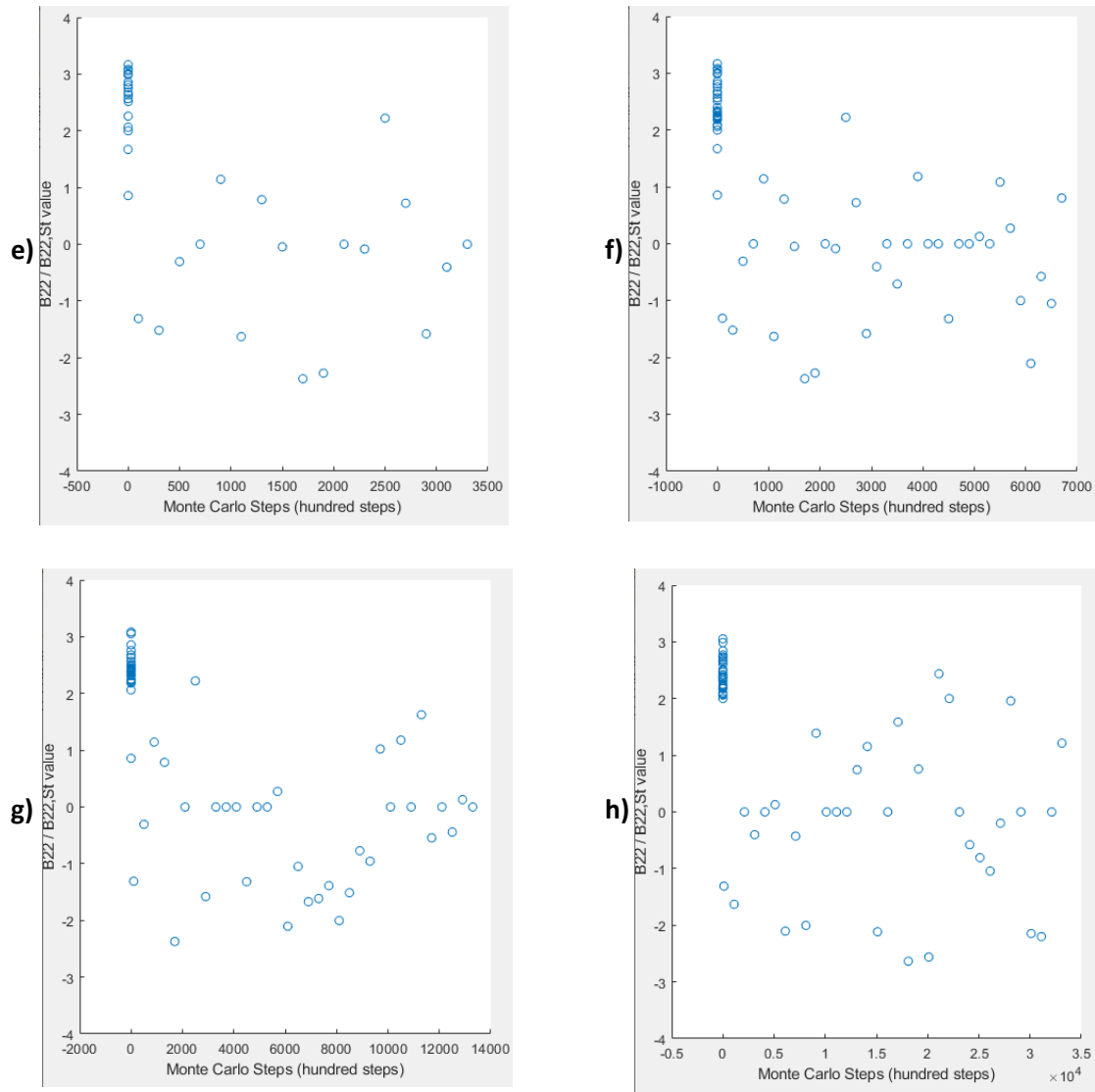
FIG. 7. The Monte Carlo simulation is run for 100 total steps and frequency varied as **a)** 0.5 **b)** 1 **c)** 2 **d)** 5.

From the results shown in FIG. 7, it is determined that the Frequency parameter controls the number of steps over which a data point is recorded. Excluding the 0.5 frequency results (it threw an error after running, stating it should be an integer), the data point density decreases for increasing frequency, as the Monte Carlo steps increase for the same. There was no expected function of this particular parameter control, as it was relatively vague in its terming. Its function is not surprising, though. It is assumed that in a real simulation, frequency could control the number of steps over which the simulation will average results to produce a data point; however, it could also be direct control of how many of the step-resultant values to exclude before recording/plotting a data point, effectively controlling the sampling frequency from the “generated” values in this “pseudo” simulation.

b) Physical Parameters

Here, the physical parameters are adjusted. The “convergency” text file being read from is changed from “convergency.txt” to “convergency1.txt” (both of which were provided). The same trials as in part a) were then conducted for the new convergency file; both sets of results are displayed below in MEGAFIG. 8.





MEGAFIG. 8. The Monte Carlo simulation is run with the new convergency file for a frequency of 1 and total steps varied as **a)** 50 **b)** 100 **c)** 200 **d)** 500. The simulation was then run for 100 total steps and frequency varied as **e)** 0.5 **f)** 1 **g)** 2 **h)** 5.

The change in convergency file appears to change the file from which the “pseudo” simulation pulls data from, in order to simulate changing conditions of the simulation. The difference in results given by the “convergency1.txt” trials is taken to represent a significant change in the continuity conditions followed to accept or reject move attempts – or a similar change within the simulation. The results in MEGAFIG. 8 support the analysis detailed in part a) as holding true. The Total Steps correlate to the amount of data points generated for a given recording frequency, while the Frequency controls over how many accepted iterations/steps a data point is recorded/plotted. This makes sense, as it matches the reasoning given in part a).

The convergency files were then inspected and directly plotted in excel [6]. They seem to be the full data sets that the “pseudo” simulation pulls sets of sampling points from. The “convergency1.txt” data set provides a not-quite-random set of values within its bounds – with a commonly recurring value of 0 in an otherwise disperse set of values. The “convergency.txt” data set appears to have been calculated with significantly better goal-sought continuity criteria.

It is important to note that Monte Carlo simulations have both benefits and drawbacks. Some benefits that Monte Carlo simulations provide are that they are less sensitive to initial conditions (as it will randomly iterate from prior moves, within the bounds of continuity criteria – effectively losing the “memory” of initial moves fairly quickly), as well as being able to simulate phase equilibrium. Some drawbacks of the Monte Carlo method are that there is no real time course, it is a poor simulation method for macromolecules within a bulk of small particles, and it is difficult to parallelize.

Thanks for a fun project!

Supplemental Information

Supplemental Information can be accessed at: <https://github.com/zswain/MSEG803>

[1] How temperature impacts pressure and potential energy:

Metrics_N=50_T=0.5_t=2E3.PNG

Metrics_N=50_T=1.0_t=2E3.PNG

Metrics_N=50_T=1.5_t=2E3.PNG

Metrics_N=50_T=2.0_t=2E3.PNG

Metrics_N=50_T=5.0_t=2E3.PNG

[2] How temperature impacts velocity distribution:

Velo_N=50_T=0.5_t=2E3.PNG

Velo_N=50_T=1.0_t=2E3.PNG

Velo_N=50_T=1.5_t=2E3.PNG

Velo_N=50_T=2.0_t=2E3.PNG

Velo_N=50_T=5.0_t=2E3.PNG

[3] How temperature impacts Total, Potential, and Kinetic Energies:

Energy_N=100_t=2E3.PNG

[4] How temperature impacts Total, Potential, and Kinetic Energies:

KinEnergy_N=100_T=0.5-Adiabatic_t=sampled.PNG

KinEnergy_N=100_T=1.0-Adiabatic_t=sampled.PNG

KinEnergy_N=100_T=2.5-Adiabatic_t=sampled.PNG

KinEnergy_N=100_T=5.0-Adiabatic_t=sampled.PNG

[5] How Potential and Kinetic Energies exchange over constant Total Energy:

Energy=5.0_N=100_T=5.0-Adiabatic_t=2E3.PNG

[6] Direct inspection of convergency files:

Matlab_convergencyFilesInspected.PNG