

Determining Free Energies of Hard Sphere Fluids via Monte Carlo Simulation

by
Austin Valeske

A PROJECT

submitted to

Oregon State University

University Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Physics and Computer Science
(Honors Scholar)

Presented 5 June 2015
Commencement June 2015

AN ABSTRACT OF THE THESIS OF

Austin Valeske for the degree of Honors Baccalaureate of Science in Physics and Computer Science presented on 5 June 2015. Title: Determining Free Energies of Hard Sphere Fluids via Monte Carlo Simulation.

Abstract approved:

David Roundy

Free energy is a fundamental property of a thermodynamic system, from which pressure, entropy, and other interesting properties can be derived. It is useful, then, to be able to accurately compute the free energy at various densities and temperatures in a way that can serve as the basis for further thermodynamic predictions.

We use Monte Carlo simulations to compute the free energy of a homogeneous hard sphere fluid, as a function of the filling fraction. We find the free energy by shrinking a valid configuration in which the component spheres are non-overlapping and checking for overlaps in the smaller volume. We develop techniques to optimize the free energy simulation for speed and accuracy, such as neighbor tables and run length tracking. Finding the absolute free energy is also discussed. This model will serve as the foundation for future simulations of more complex fluids, as free energy is an essential quantity to understand and hard spheres are the standard basis model for fluids.

We find that our free energy simulation agrees with the Carnahan-Starling equation of state, and is able to accurately predict free energy up into relatively dense states. This simulation can serve as the basis for a more complete fluid simulation that includes attractive square well interactions to model phenomena such as surface tension.

Key words: Monte Carlo Methods, Hard Sphere Fluids, Free Energy

Corresponding e-mail address: avaleske@gmail.com

© Copyright by Austin Valeske
5 June 2015
All Rights Reserved

Determining Free Energies of Hard Sphere Fluids via Monte Carlo Simulation

by
Austin Valeske

A PROJECT

submitted to

Oregon State University

University Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Physics and Computer Science
(Honors Scholar)

Presented 5 June 2015
Commencement June 2015

Honors Baccalaureate of Science in Physics and Computer Science project of Austin Valeske
presented on 5 June 2015.

APPROVED:

David Roundy, Mentor, representing Physics

Henri J. F. Jansen, Committee Member, representing Physics

Kevin McGrath, Committee Member, representing Computer Science

Toni Doolen, Dean, University Honors College

I understand that my project will become part of the permanent collection of Oregon State University, University Honors College. My signature below authorizes release of my project to any reader upon request.

Austin Valeske, Author

Table of Contents

1	Introduction	1
1.1	Motivation	1
1.2	Background	1
1.2.1	Homogeneous Hard Sphere Fluids	1
1.2.2	Carnahan-Starling	1
1.2.3	Monte-Carlo Methods	1
1.3	Theory	3
1.3.1	Helmholtz Free Energy	3
1.3.2	Free Energy from the Partition Function	3
2	Methods	4
2.1	Theory Implementation	4
2.2	Simulation Overview	4
2.3	Initialization	5
2.4	Moving to a New Configuration	5
2.4.1	Neighbor Tables	5
2.5	Testing the Small Configuration	5
2.6	Infinite Case	7
2.7	Python Script	7
2.8	Error Calculation	7
3	Results	8
3.1	Carnahan-Starling Comparison	8
3.2	Improving Simulation Accuracy	8
3.3	Triangle Test	10
3.4	Future Research	10
4	Acknowledgments	11

List of Figures

1	Calculating π with Monte Carlo methods.	2
2	Shrinking the original cell to a small cell, with no overlap.	6
3	Shrinking the original cell to a small cell. The red spheres overlap when the cell is shrunk.	6
4	The configurational free energy per sphere vs. the filling fraction, for both Carnahan-Starling and our simulation with 10 spheres. Note that, while difficult to see, there are error bars present.	9
5	The configurational free energy per sphere vs. the filling fraction, for higher numbers of spheres.	9
6	A comparison of simulation accuracy with different numbers of steps, for 5 spheres.	10
7	A “triangle test” to ensure multiple steps can be added together.	11

1 Introduction

1.1 Motivation

From the Helmholtz free energy of a system we can straightforwardly determine the entropy and internal energy of the system, and ultimately every other thermodynamic property. An accurate computational model of a system's free energy, then is extremely valuable.

The free energy of the hard sphere fluid, our specific area of interest, is already well understood[5]. We wish to develop this simulation code so it can be used as the basis for further simulations, such as for fluids of hard polyhedra. In the future, square well interactions between particles can be added to the free energy simulation to predict the surface tension and other interesting phenomena.

Code for simulating the motion of hard spheres with Monte Carlo methods (discussed in Section 1.2.3) has already been developed by Paho Lurie-Gregg and Michael Perlin. We extend this code to find the free energy of a fluid at a given density.

1.2 Background

1.2.1 Homogeneous Hard Sphere Fluids

A homogeneous hard sphere fluid has several important properties. Firstly, we simulate a homogeneous fluid using periodic boundary conditions, so spheres that exit one side of the fluid enter on the other. Secondly, hard spheres are not allowed to overlap, and in our simulation the particles are spheres for simplicity. There is code written by Lurie-Gregg that handles interacting hard polyhedra, but we opted to start with the simpler spherical case. Finally, particles in a fluid are not in any way attached to or forced into a structure with each other. They are able to move relatively freely. For this to work, we use a filling fraction η of less than about 0.7, where the filling fraction is the fraction of the volume that is filled.

1.2.2 Carnahan-Starling

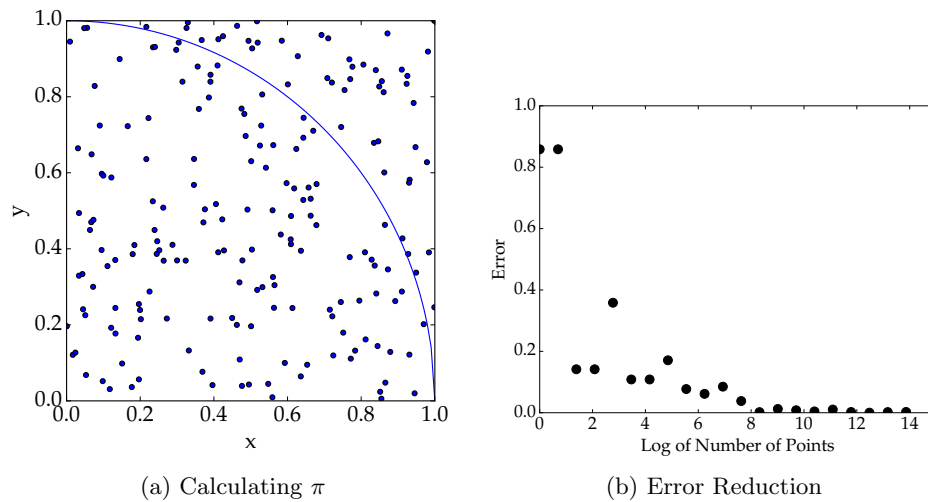
The Carnahan-Starling free energy is a well regarded approximate equation of state for a hard sphere fluid [1]. It is given by

$$\frac{F}{NkT} = \frac{4\eta - 3\eta^2}{(1 - \eta)^2} . \quad (1)$$

Monte Carlo simulations are normally used to confirm theoretical approximations, rather than the other way around, but Carnahan-Starling is sufficiently accurate that it can serve as a calibration target for our simulation.

1.2.3 Monte-Carlo Methods

A key goal of statistical mechanics is to calculate thermal averages of many-particle systems. Monte-Carlo methods are well suited for this[2], as they use repeated random

Figure 1: Calculating π with Monte Carlo methods.

sampling to model the average expected outcome of stochastic processes and processes that deal with far too many variables for analytic solutions to be practical.

A canonical example of the utility of Monte Carlo methods is to calculate the value of π . If we inscribe a circle in a unit square we know the ratio of their areas is $\pi/4$. Due to symmetry we need only take the first quadrant, as shown in Figure 1a. We then randomly add N points (of which 200 are shown) and count how many are inside the circle. This is the ratio of the two areas, so we solve $\frac{N_{\text{in circle}}}{N} = \frac{\pi}{4}$ for π . If we wish to increase the accuracy of our result we need only place more points by running the calculation longer to further probe the problem space. Figure 1b shows how the error decreases the longer we run the calculation.

Importantly, this method also allows us to calculate statistical error bars for our data. In the π calculation, each dot is like an independent experimental measurement, so uncertainty can be computed independently from statistical error. We assume a Poisson distribution, and that $N_{\text{out}} \ll N_{\text{in}}$, so that our error in N_{out} is $\sqrt{N_{\text{out}}}$. Our error in π is therefore $\delta\pi = \frac{1}{4} \frac{\sqrt{N_{\text{out}}}}{N}$.

We use this same random sampling principle in statistical mechanics to determine free energy of fluids. There is no reasonable way to analytically predict the movement of every sphere in our fluid, so instead we move each sphere randomly, checking for collisions, in a manner that resembles a random walk. After moving every sphere some number of times so our samples are sufficiently uncorrelated, we say we've reached a new configuration of the system. We can then do whatever analysis of the system we need to take a data sample. We then run the simulation further until we've reached a new configuration and take another sample. This process is repeated until we have taken enough samples that our error is sufficiently reduced.

1.3 Theory

1.3.1 Helmholtz Free Energy

Before further describing how we're using Monte Carlo methods, we need to define free energy. The Helmholtz free energy is a measure of the amount of work that can be done by a system at constant temperature. Any decrease corresponds to work being done by the system, and any increase corresponds to work being done on the system. Free energy is especially of interest to physicists because it leads directly to entropy and ultimately to all other thermodynamic quantities. Given a known system, the free energy is defined by the thermodynamic equation,

$$F = U - TS , \quad (2)$$

where we can easily know the internal energy U and the temperature T . If we also know the free energy F , then it is a simple matter to find the entropy S and thus other thermodynamic quantities of the system.

1.3.2 Free Energy from the Partition Function

Note, however, that the free energy at constant temperature can also be defined as

$$F = -kT \ln Z , \quad (3)$$

where Z is the canonical partition function. Importantly, the partition function is separable into configurational and ideal gas partition functions. Starting with

$$Z = \sum_i e^{-\beta E_i} , \quad (4)$$

we see that the energy of a sphere in the fluid is composed of the kinetic and potential energies of that sphere,

$$E = \sum_i \frac{p_i^2}{2m} + \sum_{i < j} V(R_{ij}) . \quad (5)$$

Because of this separability, we can write Equation 3 as

$$F = -kT \ln(Z^I Z^C) \quad (6)$$

$$= -kT \ln Z^I - kT \ln Z^C \quad (7)$$

$$= F^I + F^C , \quad (8)$$

where the configurational partition function Z^C is dependent on the relative positions of the particles, and the ideal gas partition function Z^I depends on the kinetic energy of the particles. As outlined in Section 2, our simulation only deals with Z^C , but Equation 7 shows that we can easily deal with the configurational free energy F^C on it's own. To

find the total free energy we need only add the ideal gas component, which we can derive analytically if needed.

Taking the F^C terms on its own, if we want to find ΔF^C between states we simply find

$$\Delta F^C = -kT \ln Z_f^C - (-kT \ln Z_i^C) \quad (9)$$

$$\Delta F^C = -kT \ln \frac{Z_f^C}{Z_i^C}. \quad (10)$$

The difference in free energy between any two states, at constant T and constant N , is simply the log of the ratio of the partition functions for the two states. In the canonical, configurational partition function for hard spheres, $Z^C = \sum_i^{\text{microstates}} e^{-\beta E_i}$, the $e^{-\beta E_i}$ term is either zero or one. The partition function, therefore, is essentially a counting of the number of micro-states in a given macro-state. This ratio is a relationship that lends itself to computational methods.

2 Methods

2.1 Theory Implementation

As outlined in Section 1.3.2, we need only find the ratio of possible configurations of particles between two different macro-states of the system. Equation 10 is valid only at constant T , and we wish to keep the number of particles N constant as well. This leaves the volume, and therefore density η , as our variable between two macro-states of the system. It is obviously impossible to count the number of possible micro-states, which in this case correspond to particle configurations, for a given macro-state, so we take a shortcut. Equation 10 tells us we only need a ratio of possible configurations between two volumes. In other words, for a configuration in a given volume, what is the likelihood that it is still valid for a smaller volume? Our simulation attempts to answer this question. Note that every microstate of the small system corresponds to a microstate of the large system, but not vice-versa.

2.2 Simulation Overview

Our process for determining F as a function of the volume consists of two programs. The computationally intensive simulation is written in C++ for speed, and the results from multiple simulations are combined by the python code described in Section 2.7.

To find ΔF^C between two volumes, the simulation goes through several steps. After initializing the system we try moving the spheres to a new, random configuration and test whether this new configuration is valid, that is, that none of the spheres are overlapping. After a sufficient number of moves have been made we try shrinking both the distances between the spheres and the dimensions of the volume by some scale factor, and check if the spheres are still non-overlapping. If any spheres overlap, the small configuration is rejected. We repeat this process several times, tracking the percentage of small configurations that were rejected. This percentage is the answer to the question we posed in Section 2.1. It gives us $\frac{Z_f^C}{Z_i^C}$, and thus the difference in configurational free energies.

2.3 Initialization

To initialize our simulation we start by defining a rectangular cuboid volume with periodic boundary conditions. Into this volume we place N spheres, where N is chosen based on the density we wish to model. We start by distributing them evenly throughout the volume on a face-centered cubic lattice, and then move the spheres until we reach some heuristically defined cutoff. Here a move is defined as in Section 2.4. As part of this process we determine the ideal translation distance, which is the distance we move a sphere in any one step.

2.4 Moving to a New Configuration

We move to a new configuration by moving every sphere some random fraction of the translation distance. As the volume has periodic boundary conditions, this could involve the sphere moving out of the volume and back in the other side. After each sphere is moved we check to make sure it is not overlapping with any other sphere, a process that is made more computationally efficient with neighbor tables, as described in Section 2.4.1. If any sphere is overlapping then that move is rejected, the sphere is not moved, and we try moving a different random sphere a different random distance. Once we have attempted to move every sphere we have completed one iteration.

This process is repeated until the new sphere configuration is sufficiently independent of the previous configuration. This is not precisely defined, but is estimated based on the number of accepted small configurations that occur in a row. If we check the small cell too often our data is overly correlated and the $O(N^2)$ expense of checking the small cell outweighs the benefit of generating new configurations. If we check the small cell too rarely we waste time generating configurations that we don't check. Our testing indicates that $\frac{N^2}{10}$ iterations is often an effective number of iterations to wait before checking a new configuration, but this varies between values of N .

2.4.1 Neighbor Tables

Moving every sphere, and checking that no spheres are overlapping after each move, is an $O(N^2)$ operation. This is extremely inefficient, especially for large N . To optimize this process we use neighbor tables. For each sphere, we store a list of the spheres in its neighbor-sphere. That is, all the spheres within some radius of the sphere's location. When we wish to make a move, we need only check for overlaps with the spheres in its neighbor-sphere, rather than all the spheres in the simulation. If a sphere moves too far from the center of its neighbor-sphere we recalculate its neighbors, as well as inform its neighbors that it has moved by adding or removing it from their neighbor tables, to keep their neighbor-spheres updated.

2.5 Testing the Small Configuration

Thus far we've been referring to shrinking the small cell as if we copy the large cell configuration, shrink the distances between the spheres and the volume's dimensions by a scaling factor, and then check for overlap. Conceptually, this is accurate, but making a

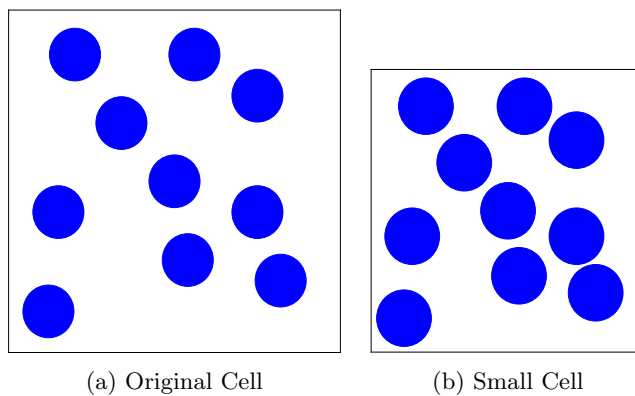


Figure 2: Shrinking the original cell to a small cell, with no overlap.

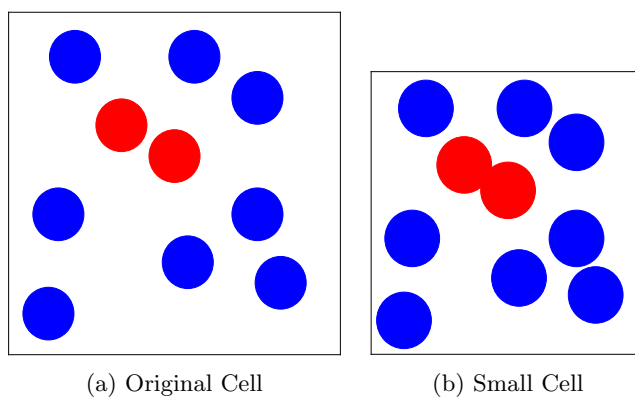


Figure 3: Shrinking the original cell to a small cell. The red spheres overlap when the cell is shrunk.

complete copy of the configuration is not necessary. Instead, when we check for overlaps by comparing the spheres' locations we first scale the spheres' position vectors by the scale factor. This is also where we ensure the sphere is still in the volume by comparing against the smaller, rescaled dimensions. This is, in effect, "shrinking the cell," even though we never generate an entire shrunken cell. With this approach the calculation is more efficient, because while checking for overlaps is in theory an $O(N^2)$ operation, in practice we rarely have to check all the spheres for overlaps. As soon as any overlap, we know the configuration is rejected and can stop.

Figure 2 illustrates what shrinking a cell looks like when the small cell is accepted, and Figure 3 illustrates the much more common case wherein the small cell is rejected. Notice that the spheres are the same size, it is only the volume and their positions relative to each other that have been reduced.

2.6 Infinite Case

The method described in Section 2.5 works well when we have a baseline free energy, but it does not work to find the absolute free energy. In order to do this we must explore the case where we move from an infinite volume to a finite volume, which is effectively the free energy required to place the spheres in an empty volume.

An infinite volume is difficult to simulate, so instead we assume that in an infinite volume the spheres will be so far apart so as to have a 0% chance of overlapping. Any large configuration is valid in the infinite case. To test the "small" volume, we simply randomly place all the spheres in our finite volume. Just as before, if any are overlapping then that small configuration is rejected. Also as before, the percentage of accepted small cell checks is the ratio of partition functions in Equation 10. By handling the infinite case, we are able to measure absolute free energies.

2.7 Python Script

Each time we run the simulation we find the ΔF for the two volumes tested in the simulation. In order to find F as a function of the volume or density, we need to run the simulation multiple times, once for each volume step we wish to take. The Python script takes care of running each step, and compiling the results into a plot like the one in Figure 4. Different numbers of spheres and different filling fractions require different steps sizes, as when there are more spheres and higher filling fractions it is less likely that the small cell will be accepted. Statistical error is also calculated in the Python code, rather than in the simulation code.

2.8 Error Calculation

As discussed when we outlined Monte Carlo methods in Section 1.2.3, we are able to compute the statistical error in our calculations. Assuming that $N_{acc} \ll N_{rej}$, where N_{acc} and N_{rej} are the number of accepted and rejected small cells, respectively, our error in N_{acc} is $\delta N_{acc} = \sqrt{N_{acc}}$.

To find the error in the free energy, we first need the error in the acceptance rate R , so that $R = \frac{N_{acc}}{N_{tot}}$ and $\delta R = \frac{\delta N_{acc}}{N_{tot}}$. From this, we take the partial derivative of $\Delta F = -kT \ln R$

so that

$$\delta\Delta F = \frac{\partial\Delta F}{\partial R}\delta R \quad (11)$$

$$= -\frac{kT}{R} \frac{\delta N_{acc}}{N_{tot}} \quad (12)$$

$$= -\frac{kT}{\sqrt{N_{acc}}} . \quad (13)$$

Because we add up multiple ΔF s to find the free energy at a given density, we must also add the uncertainties. The uncertainty for some free energy F is therefore

$$\delta F = \sqrt{\delta F_1^2 + \delta F_2^2 + \dots} . \quad (14)$$

3 Results

3.1 Carnahan-Starling Comparison

The goal of this simulation is to accurately compute the free energy of the fluid as a function of density. The Carnahan-Sterling approximation works well for low to moderate densities, so it provides a useful check on our simulation[4]. If it matches Carnahan-Starling for low densities, then it is likely that our high density simulations are accurate as well. The comparison between Carnahan-Starling and our simulation is shown in Figure 4, which shows that our simulations matches Carnahan-Starling very well under the right circumstances. In this case, we ran the simulation with 1,000,000 iterations, and 10 iterations between each check of the small cell.

3.2 Improving Simulation Accuracy

Figures 5 and 6 demonstrate various issues that come up while running simulations that should be avoided. Figure 5 shows the simulation run with 20 spheres (5a) and 50 spheres (5b). In both of these simulations, while the results themselves fit very closely to the Carnahan-Starling curve, the error bars are extremely large. As shown in Equation 13, the size of the error bars is entirely dependent on the number of accepted small cells, with the additional constraint that $N_{acc} \ll N_{tot}$. While these simulations also ran for 1,000,000 iterations, the larger number of spheres means they not only checked the small cell less frequently, but found fewer valid small cells when they did. This resulted in fewer accepted small cells and a larger error. To reduce this error, we need only run the simulation longer.

Of additional interest is the compounding nature of statistical error. Figure 5a clearly shows how the error grows as additional steps are taken. This is partially due to higher values of η resulting in fewer accepted small cells, but it is also a result of the way the error builds on itself, as shown in Equation 14. This can also be compensated for by running the simulation longer and avoiding taking unnecessarily small steps.

Finally, Figure 6 shows how important it is to take an appropriate number of steps. When too few steps are taken, as in Figure 6a, the predicted free energies are nowhere near those estimated by Carnahan-Starling. With a more reasonable number of steps, as in 6b, the predicted free energies better fit Carnahan-Starling's estimates.

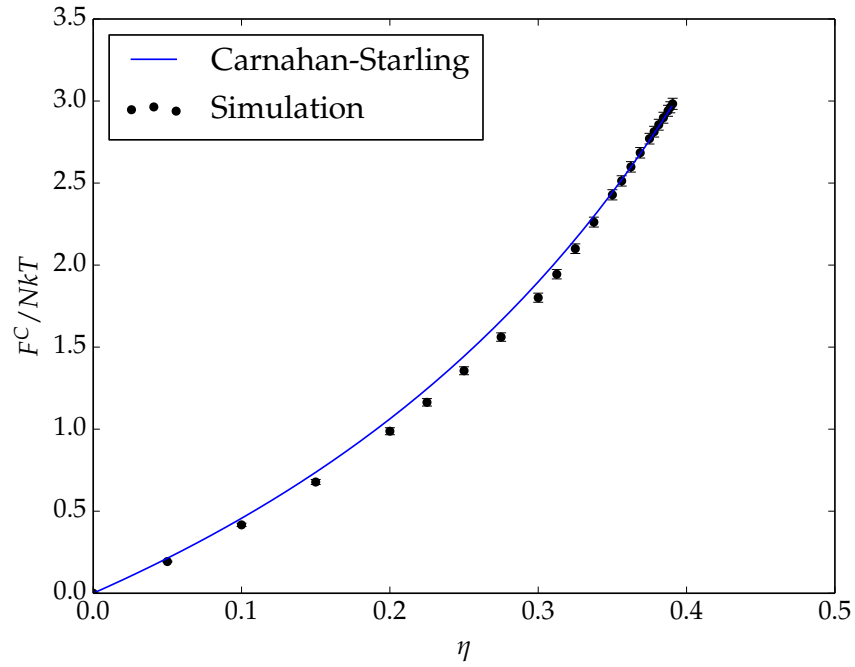


Figure 4: The configurational free energy per sphere vs. the filling fraction, for both Carnahan-Starling and our simulation with 10 spheres. Note that, while difficult to see, there are error bars present.

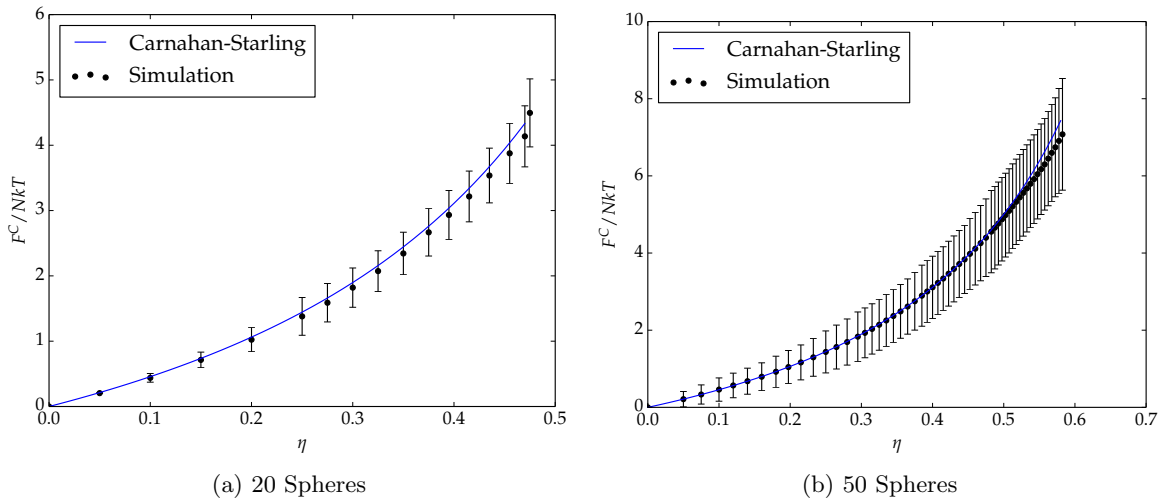


Figure 5: The configurational free energy per sphere vs. the filling fraction, for higher numbers of spheres.

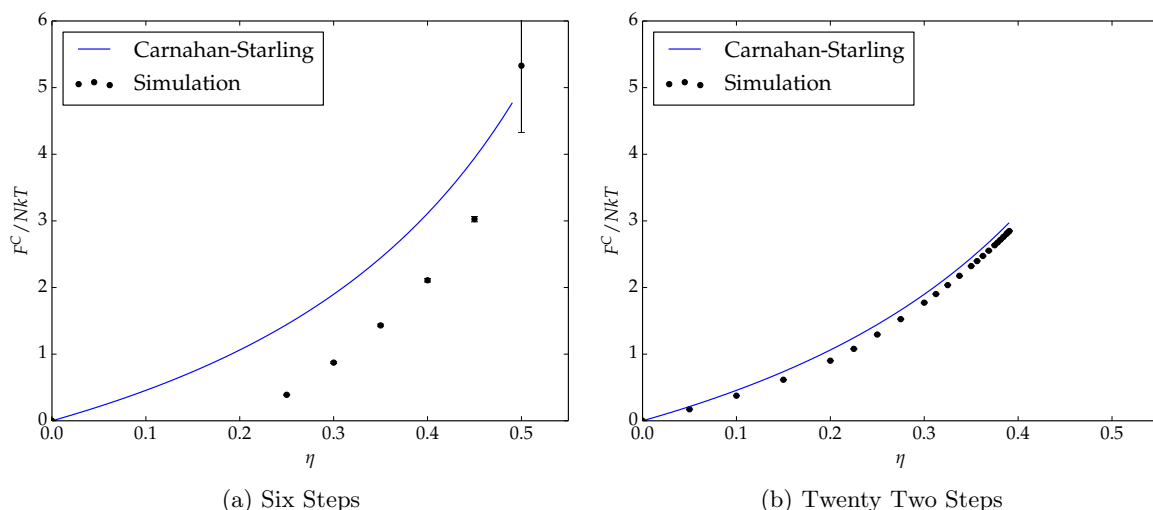


Figure 6: A comparison of simulation accuracy with different numbers of steps, for 5 spheres.

3.3 Triangle Test

In order to ensure our simulation is accurate, we also performed a basic “triangle test”. As shown in Figure 7, the triangle test consists of moving between two points in our problem space, but along two different paths. On the first path we move from $\eta = 0.3$ to $\eta = 0.33$ in one step, and on the second path we take two steps. This test is used to ensure accuracy and tune the simulation. Preliminary testing with 10 spheres and 1,000,000 iterations indicates that our simulation is accurate to within our expected error.

3.4 Future Research

While hard sphere and polyhedral fluids are not complex enough to model surface tension and other phenomena, they do allow simulations of the freezing transition and provide an ideal starting point [3]. We now have a working free energy model for hard sphere fluids. From here we can move to polyhedral fluids and incorporate square well interactions from the model currently under development by Perlin. These additional components will allow modeling of interesting phenomena, such as surface tension and fluid behavior near the critical point.

The simulation results shown in Figures 4 through 6 are plotted into the highest densities for which they were functional. In theory we should be able to compute free energies with higher η values than those shown, but our initialization routine prevents this. As noted in Section 2.3, the spheres are placed on a face-centered cubic lattice when they are first placed in the volume. This is not dense enough for higher density states, and the initialization routine fails. Modifications to the initialization routine should allow modeling of phenomena such as freezing.

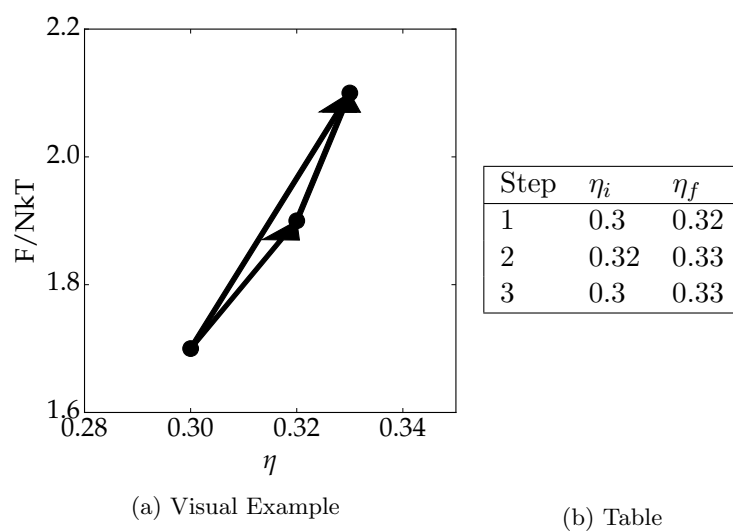


Figure 7: A “triangle test” to ensure multiple steps can be added together.

4 Acknowledgments

Michael Perlin, Paho Lurie-Gregg, and David Roundy were very helpful in understanding current research and the tools that were needed for this project.

References

- [1] Norman F. Carnahan and Kenneth E. Starling. Equation of state for nonattracting rigid spheres. The Journal of Chemical Physics, 51(2):635–636, 1969.
- [2] D.P. Landau and K. Binder. A Guide to Monte Carlo Simulations in Statistical Physics. Cambridge University Press, 2005.
- [3] Paho Lurie-Gregg. The contact value approximation to the pair distribution function for an inhomogeneous hard sphere fluid. <http://hdl.handle.net/1957/54549>, 2014.
- [4] Yuhua Song, E. A. Mason, and Richard M. Stratt. Why does the carnahan-starling equation work so well? The Journal of Physical Chemistry, 93(19):6916–6919, 1989.
- [5] Martin B Sweatman. Comparison of absolute free energy calculation methods for fluids and solids. Molecular Physics, (ahead-of-print):1–11, 2015.