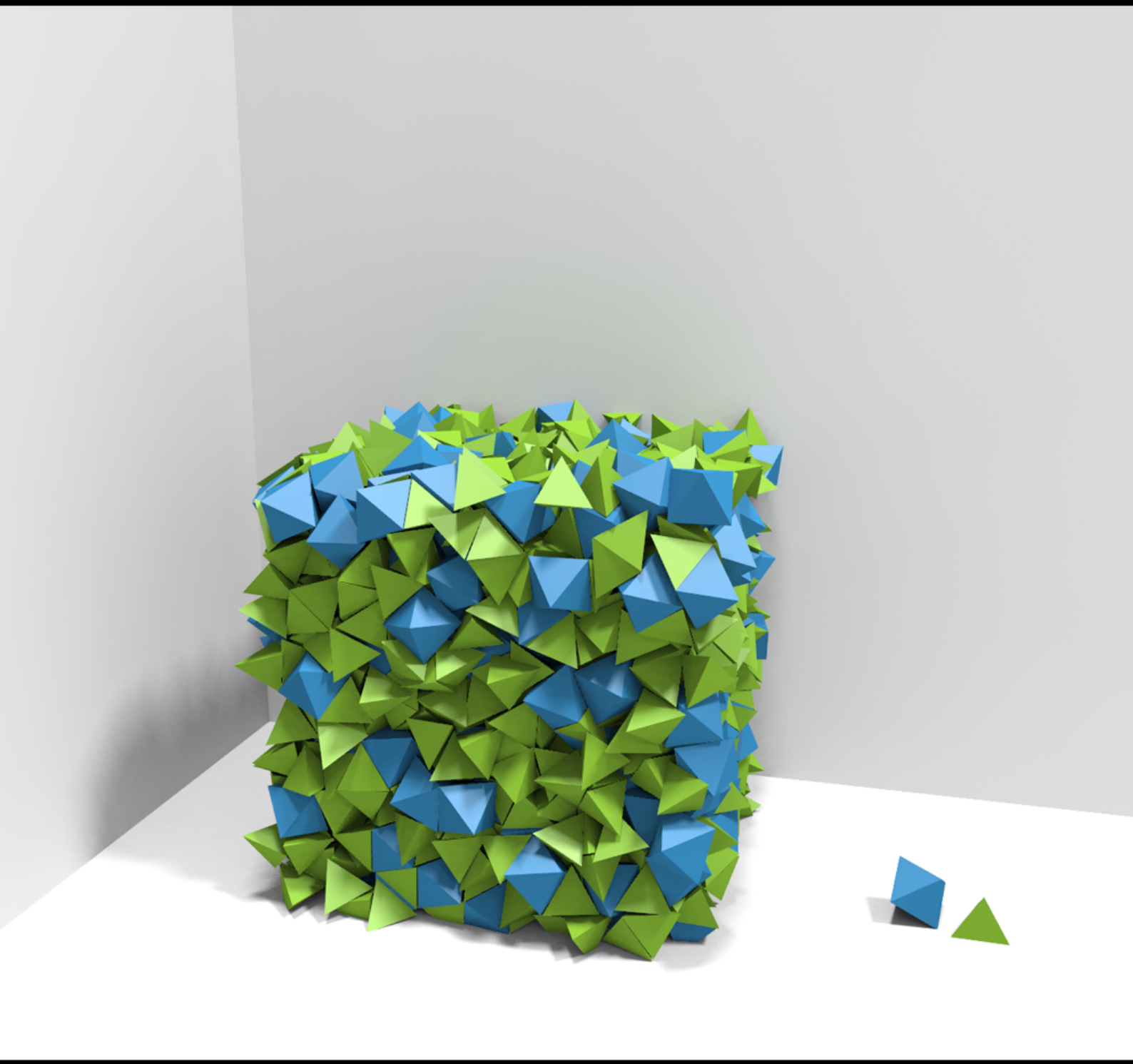
Master Thesis - University Utrecht



Glassy Dynamics of Convex Polyhedra

Nick Tasios

Glassy Dynamics of Convex Polyhedra

Nick Tasios

Master Thesis Theoretical Physics
February 2012 - January 2013
Debye Institute for Nanomaterial(s) Science
Soft Condensed Matter Group

Department of Physics and Astronomy
Faculty of Science
Utrecht University

Supervisors:
Anjan Gantapara
René van Roij
Marjolein Dijkstra

Dedicated to my little sister.

Acknowledgments

I would like to thank Marjolein Dijkstra and Anjan Gantapara for their patience and continuous help in the realization of this Thesis. I would also like to thank Djamel el Masri for sharing his knowledge of calculating the SISF.

Abstract

Self-assembly of polyhedral-shaped particles have attracted huge interest with the advent of new synthesis methods to realize these faceted particles in the lab. Recent studies have shown that polyhedral particles exhibit rich phase behavior by excluded volume interactions alone. Some of these particles are alleged to show a transition to a glass phase by quenching the liquid sufficiently fast beyond the glass transition (supercooling) such that the formation of structures with long-range order is suppressed. Despite the recent progress, no study has been made on the glass formation of polyhedral-shaped particles. Here we study the glassy behavior of polyhedral particles using advanced Monte Carlo methods. In the first part of this thesis the formation of glass by mono-disperse polyhedral particles, namely octahedra, tetrahedra and triangular cupola are discussed. At the end, the fragility of these particles is calculated and compared with that of other systems.

In the second part of this thesis we performed preliminary calculations on binary mixtures of polyhedral particles. Possible candidate crystal structures are studied using the floppy box Monte Carlo method.

Contents

1

Introduction

1.1 Glasses

In layman's terms, the glass phase is an amorphous solid phase. That is, no long-range order is present but there may be short range order (fig. 1). The most familiar type of glass is the one used in windows, drinking vessels etc. In reality, this metastable state has many definitions and many ways by which one can identify it and contrary to what one might think, glasses are not at all well understood. Beside the standard glass, many materials can form glasses with different kinds of properties. Generally, of interest is creating strong and low-density glasses and glasses with various optical and electrical properties. A recent hot topic is also what is called a metallic glass or an amorphous metal because of its toughness and small fragility, the latter as a consequence of the absence of crystal defects.

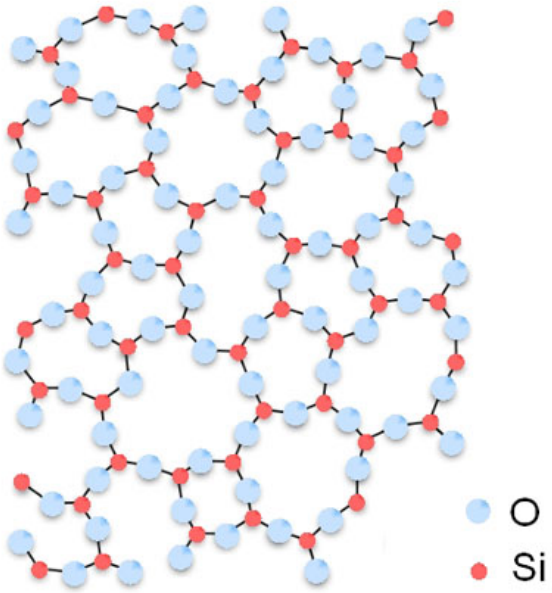


Figure 1: The amorphous structure of glassy Silica (SiO2) in two dimensions. No long range order is present, however there is local ordering with respect to the tetrahedral arrangement of oxygen (O) atoms around the silicon (Si) atoms.

The scientific community has shown great interest in glasses because the nature of the glass transition is still not well understood and is under continuous debate up to today. Materials close to the glass transition show a critical slowing down of their dynamics, which coupled with the fact that it lacks any long range order, allows us to think of a glass as a liquid with a very high viscosity. Because of the slow dynamics, it is hard to study the long time behavior of glasses. The glass transition can either occur upon cooling or compression and is accompanied by a smooth increase in the viscosity (unlike first-order phase transitions where abrupt changes are involved). The viscosity of a material can increase by as much as

17 orders of magnitude without any pronounced change in material structure. Whether the glass transition is a thermodynamic or kinetic phase transition, is a hot topic of debate. A common method to create a glass from a glass forming liquid is to supercool it. Supercooling, also known as undercooling, is the process of lowering the temperature of a liquid or a gas below its freezing point without becoming a solid. A liquid below its standard freezing point will crystallize in the presence of a seed crystal or nucleus around which a crystal structure can form. However, in the absence of such a nucleus, the liquid phase can be maintained all the way down to the temperature where homogeneous crystal nucleation occurs. If the system is quenched sufficiently fast such that homogeneous nucleation has not occured, an amorphous (non-crystalline) solid will form.

The dynamics of supercooled liquids and the related phenomenon of the glass transition has been the focus of interest for a long time [1]. The reason for this is the fact, that if a glass former is cooled from its melting temperature to its glass transition temperature T_g , it shows an increase of its relaxation time by up to 14 decades without a significant change in its structural properties, which in turn poses a formidable and exciting challenge to investigate such a system experimentally as well as theoretically. Remarkably, supercooled liquids show a very similar scaling behavior for the relaxation times for a very large variety of different liquids. Despite the efforts to study this dramatic growth in relaxation time, even today there is still intense dispute on what the actual underlying mechanisms for this increase is. Over the course of time many different theories have been put forward, such as, the entropy theory by Adams, Gibbs and Di Marzio [2, 3], the coupling-model proposed by Ngai [4], or the mode-coupling theory (MCT) by Götze and Sjögren [5, 6]. The approach by which these theories explain the slowing down of the supercooled liquid with decreasing temperature differs radically from case to case. In the entropy theory it is assumed e.g., that the slowing down can be understood essentially from the thermodynamics of the system, whereas MCT puts forward the idea that at low temperatures the nonlinear feedback mechanisms in the microscopic dynamics of the particles become so strong that they lead to structural arrest of the system. The most successful of the theories has been MCT, which makes far more predictions that corroborate with experiment than the other theories. We will present some of these predictions and show how well they agree with our results later on.

1.2 Convex Polyhedra

The formation of a glass depends primarily on the local ordering. Movement of a partile can be confined by the cage formed by the surrounding particles. As such, it is usually believed that some kind of anisotropy needs to be present in the particles for them to form a glass. As our study shows, even particles exhibiting large symmetry can form glasses.

A convex polyhedron can be defined algebraically as the set of solutions to a system of linear inequalities

$$M\mathbf{x} \leq \mathbf{b}$$

where M is a real $s \times 3$ matrix and $\mathbf{b}$ is a real s-vector. Geometrically, a convex polyhedron can be defined as a polyhedron for which a line connecting any two (non-coplanar) points on the surface, always lies in the interior of the polyhedron. The 92 convex polyhedra having only regular polygons as faces are called the Johnson solids, which include the Platonic solids and Archimedean solids. No method is known for computing the volume of a general convex polyhedron [7]. Some examples of convex polyhedral-shaped nanoparticles are shown in figs. 2 and 3.

In this thesis, we investigated the glassy behavior of a few of these nanoparticles. We assume the polyhedra studied, to interact solely through a hard-core potential i.e. they can not overlap. This excluded volume interaction is sufficient for a large diversity in structures although not as diverse as other systems with more complex interactions. It is though interesting to see how shape alone can have such an influence in how various materials self-assemble in an entropy driven process.

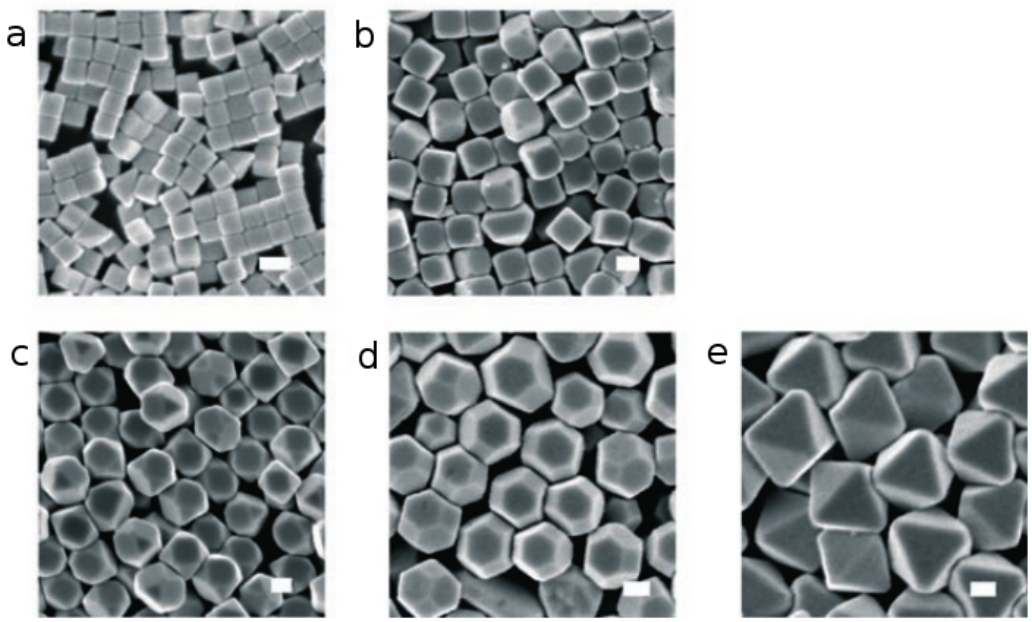


Figure 2: SEM images of a) cubes, b)truncated cubes, c) cuboctahedra, d) truncated octahedra and e) octahedra [8].

Lately, there has been an increasing interest in the production and self-assembly of polyhedral particles [9–15]. Much progress has been made in enumerating and characterizing the packing of polyhedral shapes and the self-assembly of polyhedral nanocrystals into ordered superstructures [16]. New synthetic methods give access to a wide range of well-defined polyhedral nanocrystalline shapes and faceted nano-particles including cubes, truncated cubes, cuboctahedra, truncated octahedra and octahedra over a range of sizes from 100 to 300 nm [8, 17–25]. A few such polyhedral nanoparticles can be seen in fig. 2. Some ordered nanostructures formed by polyhedral particles are shown in fig. 3.

The experimental realization of these kind of particles has sparked a driving interest in their dynamics and self-assembly [26–28]. Glotzer et al. [26] studied the self-assembly of 145 convex polyhedra and also found a quasi-crystalline phase of hard tetrahedra [29]. Very recently some advancement has also been made in the research of binary compounds of polyhedra particles [30].

1.3 Computer Simulations

In a computer simulation one tries to mimic the behavior of a system or process. Computer simulations are used in many disciplines such as natural sciences, economics, social sciences etc. Computer simulations can give insight on new and even known phenomena due to their

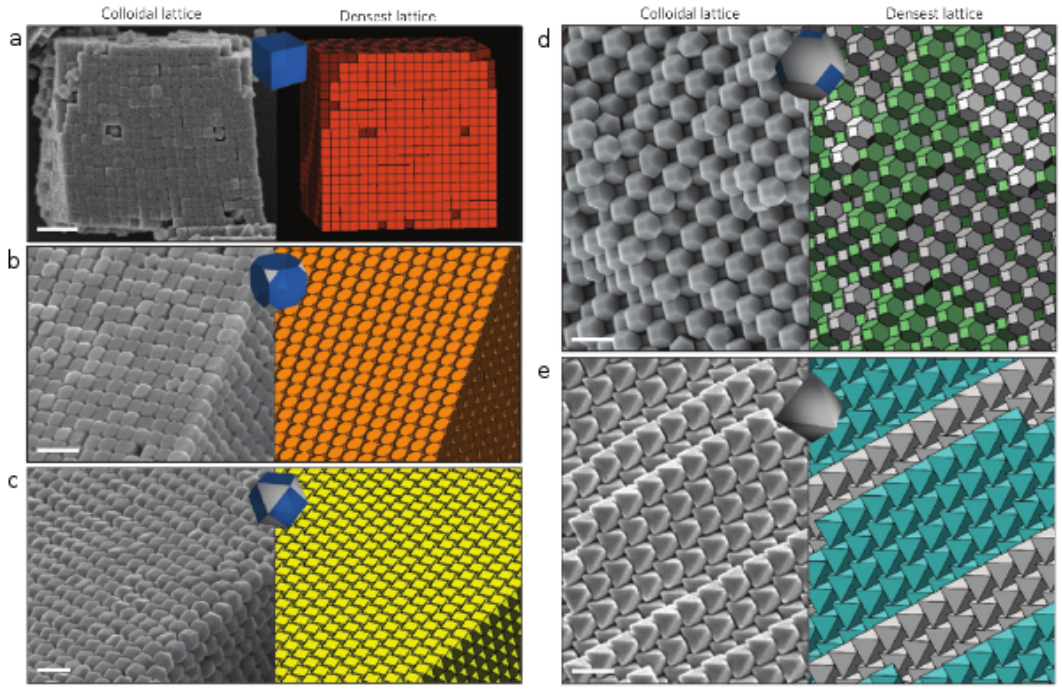


Figure 3: Self-assembly of polyhedral nanoparticles into dense lattices. SEM micrographs of the colloidal crystal lattices (left) and the corresponding diagrams of their densest known lattice packings (right): **a**, cubes; **b**, truncated cubes; **c**, cuboctahedra; **d**, truncated octahedra; **e**, octahedra. Colour in **d** and **e** is used to indicate different layers of the crystal. Scale bars are 500 nm unless otherwise noted [9].

flexibility compared to experiments. The first computer simulations were conducted nearly as soon as the first computers were invented and their development went hand-in-hand with that of computers. The first large scale deployment was during the Manhattan Project in World War II to model the process of nuclear detonation. It was a simulation of 12 hard spheres using a Monte Carlo algorithm.

Two main properties which characterize computer simulations, are whether a simulation is stochastic or deterministic and whether they are continuous or discrete. In this thesis, we use "continuous" stochastic simulations, where the stochastic element is introduced via pseudo-random number generators. Of particular interest are a class of stochastic computer algorithms, called Monte Carlo methods. The Monte Carlo method was developed by John von Neumann and Stanislaw Ulam in 1946. The name refers to the Monte Carlo casino in Monaco where Ulam's uncle borrowed money to gamble. The benefits of the Monte Carlo method over other non-stochastic methods is that it can be much faster both in performance and in drawing conclusions because not all possibilities need to be considered. Monte Carlo is usually accompanied by the use of statistics to interpret the results.

Since computers have a finite power and also due to Moore's law, only a finite part of the particular model we're trying to simulate can be actually simulated. For example, in our simulations, we typically use 2000 particles. This is a big improvement over the handful of particles used 50-60 years ago. Due to this limitation we usually use come kind of boundary conditions to make the simulation space look infinite, but that in turn may introduce other problems which are called finite-size effects. The larger the system that is simulated the less these effects affect the outcome.

2

Theory and Methodology

2.1 Statistical Mechanics

Statistical Mechanics is primarily concerned with the calculation of properties of condensed matter systems. These systems are usually composed by a great number of particles, i.e. the Earth's atmosphere contains about 4×10^{44} molecules. For example, to calculate the pressure exerted by a mole of hydrogen gas, which contains an Avagadro number of atoms ($\sim 6 \cdot 10^{23}$), on its container one needs to solve the equations of motion for Avagadro number of molecules. As one might imagine, solving an exact many-body equation for such a large system is impossible. Statistical mechanics tries to avoid solving the equations of motion by treating the system as a probabilistic ensemble and calculating the properties of interest as ensemble averages. For such large systems, probabilities are nearly as good as facts hence giving us the ability to calculate these properties within very good approximation.

The systems studied by statistical mechanics are usually governed by a Hamiltonian function H which gives the total energy of the system in any state. For a system to traverse the various available states it needs to be stimulated from an outside source, such source is called a thermal reservoir. A thermal reservoir is an external system which acts as a source and sink of heat which constantly exchanges energy with the system under study. The thermal reservoir can be incorporated as a perturbation in the Hamiltonian.

Suppose $R(\mu \to \nu)dt$ is the probability a system in state μ is found in a state ν dt time later. $R(\mu \to \nu)$ is the transition rate from state μ to state ν. It is usually assumed that the transition rate is time independent which we will also assume here. If we define $w_\mu(t)$ as the set of probabilities for the system being in state μ at time t, we may write a master equation for the evolution of these probabilities:

$$\frac{dw_\mu(t)}{dt} = \sum_\nu [w_\nu(t)R(\nu \to \mu) - w_\mu(t)R(\mu \to \nu)]$$

which may be translated as, the evolution of $w_\mu(t)$ is equal to the sum of the probabilities of the system going into state μ minus the probabilities the system is going out of that state. The sum of the probabilities $w_\mu(t)$ should obey:

$$\sum_\mu w_\mu(t) = 1$$

since the system should always be in some state. These weights are important in figuring out related macroscopic quantities. For example, for a thermodynamic quantity Q, which takes the value Q_μ in state μ, the expected value is equal to:

$$\langle Q \rangle = \sum_\mu Q_\mu w_\mu(t)$$

As the name implies, $\langle Q \rangle$ is the expected value (the most probable) and not the observed one if one was to make an instantaneous measurement of the system. It may be thought as the most probable value if we were to make many observations.

An important role in statistical mechanics is played by the partition function Z, which describes the statistical properties of a system in thermodynamic equilibrium. There are several different types of partition functions corresponding to different types of statistical ensembles. For instance, the partition function for the "canonical ensemble" is:

$$Z = \sum_{\mu} e^{-\beta E_\mu}$$

where β is the inverse temperature and E_μ is the energy of state μ. It is interesting to note that the equilibrium values of the weights $w_\mu(t)$ are called equilibrium occupation probabilities and are denoted by:

$$p_\mu = \lim_{t\to\infty} w_\mu(t)$$

which are in turn connected to the partition function by:

$$p_\mu = \frac{1}{Z} e^{-\beta E_\mu}$$

also known as the Boltzmann distribution. From the above, we can see that we may write for the expected value $\langle Q \rangle$ of a system in equilibrium, we have:

$$\langle Q \rangle = \sum_{\mu} Q_\mu p_\mu = \frac{1}{Z} \sum_{\mu} Q_\mu e^{-\beta E_\mu}$$

2.2 Monte Carlo and the Metropolis algorithm

As we saw in the previous subsection, to find the expectation value $\langle Q \rangle$ of a thermodynamic quantity Q, we average over all states weighing each with it's Boltzmann probability. For very small systems with a small number of states, this is feasible, but for larger systems, which are usually of interest, this becomes very difficult, if not impossible. Thus, in reality we choose a subset of all states at random, with some probability p_μ. The question then is how to chose the probabilities p_μ such that we get an accurate estimate of $\langle Q \rangle$. Given M states, the estimator Q_M of $\langle Q \rangle$ is:

$$Q_M = \frac{\sum_{i=1}^{M} p_{\mu_i}^{-1} Q_{\mu_i} e^{-\beta E_{\mu_i}}}{\sum_{j=1}^{M} p_{\mu_i}^{-1} e^{-\beta E_{\mu_i}}} \tag{1}$$

The above expression becomes exact when $M \to \infty$. The simplest choice for the probabilities p_μ is to make them all equal but in certain circumstances it may be give a poor estimation of $\langle Q \rangle$. The problem is that the sums in eq. (1) may be dominated by a small number of states with all other states contributing very little. To sample instead just the important states and thus get a good estimate we use the technique called importance sampling.

The strategy of importance sampling is to pick the M states described above with a probability equal to the Boltzmann distribution, $p_\mu = \frac{1}{Z} e^{-\beta E_\mu}$. The estimator Q_M then becomes just:

$$Q_M = \frac{1}{M} \sum_{i=1}^{M} Q_{M_i} \tag{2}$$

This definition works much better than the previous one, especially when the system spends the most time in a small number of states since these will be precisely the states we most often

and the relative frequency with which we pick them will exactly correspond to the amount of time the real system would spend in them. It now remains to see how we pick our states such that they have the correct Boltzmann probability.

Almost all Monte Carlo schemes rely on Markov processes to generate a state ν given a state μ. The probability of generating state μ given state ν is called the transition probability $P(\mu \to \nu)$ and for all Markov processes, it should satisfy two conditions. The probabilities should not vary in time and they should not depend on the system's history but only on the states μ and ν. The transition probabilities should also satisfy:

$$\sum_{\nu} P(\mu \to \nu) = 1$$

since some state must be generated. The Markov process is usually chosen such that given enough time, the states will follow the Boltzmann distribution at which point it will have reached equilibrium or stationary distribution. We additionally place two more conditions to achieve this, called "ergodicity" and "detailed balance".

Ergodicity is the requirement that it should be possible to reach any state from any other state in a finite amount of time. This condition is needed to ensure that all states are generetated with the correct Boltzmann distributions. Detailed balance on the other hand is a condition which ensures the correct stationary distribution is reached. It is described as the requirement that the probability of making a transition from one state to an other one is the same as the probability for the reverse transition. It should be noted that although the detailed balance conditions is sufficient, it is not a necessary condition as shown e.g. in [31].

For a more concise and detailed description of the Monte Carlo method we recommend reading [32].

2.3 Hard Core Potential Simulations in the NVT and NPT ensembles

2.3.1 Basis of the simulations

When dealing with Monte Carlo simulations in the NVT or NPT (and other) ensembles, the general approach is to put the particles in a box and perform trial moves at each Monte Carlo step. A variety of different kinds of trial moves are available depending on the ensemble and particle characteristics. The most basic trial moves for the NVT ensemble are a translational move and an orientational move (for spheres, such move is not needed). The NPT ensemble has an additional volume move in which the box gets contracted or retracted. If one wishes to study equilibrium crystal structures, an additional sheer move of the box is recommended [33]. This addition of the sheer move is often referred to as the floppy box method. More details follow in section 2.3.2.

As mentioned above, in the NVT ensemble we perform either a trial translational or orientational move. A Monte Carlo step of a basic NVT simulation can be thus summarized as follows [34]:

1. Select a particle at random and calculate the energy of this configuration $U(o)$.
2. Either give this particle a random displacement,

   $$\vec{r}(o) \to \vec{r}(o) + \vec{\Delta x}$$

   ,or randomly change the orientation of the particle. The details of the orientational move will be discussed later.

3. Calculate the energy of the new configuration $U(n)$ and accept it with a probability,

$$P(o \to n) = min\left(1, exp\left[-\beta\left(U(n) - U(o)\right)\right]\right)$$

if rejected, the old configuration is kept.

We choose either the orientational or the translational moves with a probability of 0.5 and one Monte Carlo move is defined as N such trials, where N is the total number of particles. $\vec{\Delta x}$ is a random vector inside a cube of edge length dx, where dx is chosen such the acceptance rate of the translational moves stays between 25% and 30%. For the Hard Core Potential we have to additionally check for overlaps between particles in step 3 above before accepting the new configuration. The Hard Core potential is defined as,

$$V(r) = \begin{cases} 0, & r < r_{max} \\ \infty, & r < r_{max} \end{cases}$$

where r_{max} is the distance between the two particle centers, when their boundaries are in touch.

2.3.2 Volume moves in the NPT ensemble and the floppy-box method

As mentioned before, for the NPT ensemble we have an additional volume move or a scaling of the box and the positions of the particles. One such trial move is attempted in every Monte Carlo step and thus a Monte Carlo step is defined as $N + 1$ trials. In our simulations this volume move is described as,

$$V(o) \to V(o) + \Delta V$$

and the new configuration is accepted, before checking for collisions between particles, with probability,

$$P(o \to n) = min\left(1, exp\left[-\beta\left(U(n) - U(o) + P(V(n) - V(o)) - N\beta^{-1}\ln(V(n)/V(o))\right)\right]\right)$$

ΔV is chosen randomly in the range $(0, V_{max}]$ and V_{max} is further chosen such that the trial move has a reasonable chance of being accepted. We change V_{max} every 1000 Monte Carlo steps such that the acceptance of a volume move stays between 10% and 15%. The box is scaled in each of the three directions along with the particle coordinates by

$$L'_{x,y,z} = \frac{V(n)}{V(o)} L_{x,y,z} = \frac{V(o) + \Delta V}{V(o)} L_{x,y,z}$$

If one further desires to add box deformations to the trial moves, one has to use a matrix representation of the box. It is easy to create such a matrix, if $\vec{a}$, $\vec{b}$, $\vec{c}$ are the three vectors along the edges of the box with a vertex as a common origin, then we can create a 3×3 matrix

$$B = \begin{bmatrix} \vec{a} & \vec{b} & \vec{c} \end{bmatrix} = \begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix}$$

where the diagonal is responsible for scaling and off-diagonal elements are responsible for deformations. To avoid rotating the box without changing any of it's properties, we usually choose to set the lower triangular part of the matrix to zero. This method is also called the floppy box method. For small number of particles and large deformations, due to the boundary conditions imposed on the walls of the simulations box, additional collision checks with the particles' n-periodic images are needed. We can get the n-periodic images by translating the particle position $\vec{r}$ by $n_1\vec{a} + n_2\vec{b} + n_3\vec{c}$, where n_1, n_2, n_3 are integers.

2.3.3 Particle Orientation, Quaternions and the Triad Method

There are many different representations of orientation, these include the famous Euler angles, quaternions, the rotation matrix representation etc. Some of these representations can suffer from what is called gimbal lock which is the loss of one degree of freedom in a three-dimensional space that occurs when the axes of two of the three gimbals are driven into a parallel configuration, "locking" the system into rotation in a degenerate two-dimensional space. Representations such as the Euler angles are easier to work with mentally, but we chose to work with quaternions instead because of their efficiency and the fact it's easier to uniformly sample random rotations.

Quaternions can be thought of as an extension to complex numbers, they have a real part and a complex vector

$$q = (r, \vec{v})$$

where $\vec{v} = ai + bj + ck$ and $i^2 = j^2 = k^2 = ijk = -1$. The formulas for addition and multiplication are,

$$q_1 + q_2 = (r_1, \vec{v}_1) + (r_2, \vec{v}_2) = (r_1 + r_2, \vec{v}_1 + \vec{v}_2)$$

$$q_1 q_2 = (r_1, \vec{v}_1)(r_2, \vec{v}_2) = (r_1 r_2 - \vec{v}_1 \cdot \vec{v}_2, r_1\vec{v}_2 + r_2\vec{v}_1 + \vec{v}_2 \times \vec{v}_2)$$

Unlike multiplication of real or complex numbers, multiplication of quaternions is not commutative. Similarly to the Euler formula, a quaternion rotation can be constructed using the formula:

$$q = e^{\frac{1}{2}\theta(ai+bj+ck)} = \cos\frac{1}{2}\theta + (ai + bj + ck)\sin\frac{1}{2}\theta$$

where θ is the angle of rotation and (a, b, c) is the axis of rotation. If we now define the quaternion,

$$q = \cos\frac{1}{2}\theta + \vec{u}\sin\frac{1}{2}\theta$$

where $\vec{u}$ is a unit vector, it can be shown that for a ordinary vector $\vec{v}$, it can be rotated using the following identity:

$$\vec{v}' = q\vec{v}q^*$$

where q^* is the quaternion conjugate, analogue of the complex conjugate. $\vec{v}$ can be also thought of as a quaternion with zero real part. Since we're also interested in computational efficiency of our rotation operation, we use the following formula to calculate the new orientation $\vec{v}_{new}$ of a particle:

$$\vec{v}_{new} = \vec{v} + 2\vec{q} \times (\vec{q} \times \vec{v} + w\vec{v})$$

where the quaternion $q = (w, \vec{q})$.

As we described earlier, we want to produce orientational trial moves similar to what was done for the translational moves. We thus need a method to generate random quaternions given a control angle θ_{max} which is the maximum random angle of the rotation. Generating a uniform distribution is not as straightforward as one might think. The problem of generating a random vector on a unit hypersphere is called sphere point picking. We choose to generate a random rotation in the axis-angle representation (i.e. $(\vec{v}, \theta)$, where $\vec{v}$ is the axis and θ is the angle) and then convert the resulting rotation to a quaternion using,

$$q = (\cos\theta/2, \vec{v}\sin\theta/2)$$

Marsaglia [35] derived an elegant method that consists of picking x_1 and x_2 from independent uniform distributions on $(-1, 1)$ and rejecting points for which $x_1^2 + x_2^2 \geq 1$. From the remaining points,

$$x = 2x_1\sqrt{1 - x_1^2 - x_2^2}$$
$$y = 2x_2\sqrt{1 - x_1^2 - x_2^2}$$
$$z = 1 - 2\left(x_1^2 + x_2^2\right)$$

have a uniform distribution on the unit sphere.

For a uniform distribution, the distribution function of the unit vector $\vec{\mathbf{n}}$ should be equal to,

$$p(\vec{\mathbf{n}}) = \frac{1}{S_3} = \frac{1}{2\pi^2}$$

Let Ω represent a patch of area on the unit sphere. Then the probability that the unit normal will lie within that patch is

$$\int_\Omega p_{\vec{\mathbf{n}}}(\vec{\mathbf{n}}) = \frac{1}{2\pi^2}\int_\Omega dA$$

where $dA = \sin^2\phi_1 \sin\phi_2 d\phi_1 d\phi_2 d\phi_3$. The distribution function for ϕ_1 is defined as,

$$\int_{\phi_a}^{\phi_b} p_{\phi_1} d\phi_1 = \frac{1}{2\pi^2}\int_{\phi_a}^{\phi_b} \sin^2\phi_1 d\phi_1 \int_0^\pi \int_0^{2\pi} \sin\phi_2 d\phi_2 d\phi_3 =$$
$$= \frac{2}{\pi}\int_{\phi_a}^{\phi_b} \sin^2\phi_1 d\phi_1$$

Thus, the probability density p_{ϕ_1} is proportional to $\sin^2\phi_1$. We can then use rejection sampling to sample this density using the density $g(a) = a^3$. This is done by sampling an angle in the range $[0, \theta^3)$ and accepting the result with probability $\sin^2 a/a^2$ (which should be close to one for small enough angles). Notice that we only need to sample positive angles since by symmetry, opposite axes, give us opposite rotation angles.

When rotating particles in a simulation, one needs to keep track of their orientation. This can be done by doing an additional quaternion-quaternion multiplication every time a orientational trial move has been accepted. This can be costly for both the CPU and memory, especially if we only need to know the orientation every few simulation steps. What can be done instead is to keep track of the particle vertices' positions w.r.t. the particle's center and then extract the orientation. The method used to extract this information is called the Triad Method. Given two linear independent reference vectors $\vec{r_1}, \vec{r_2}$ and the two rotated vectors in the current frame of interest $\vec{R_1}, \vec{R_2}$ the Triad Method can find a rotation matrix A which takes the system from one frame to the other. The method proposes an estimate of the direction cosine matrix A as a solution to the linear system of equations given by,

$$\left[\vec{R_1}\vdots\vec{R_2}\vdots\left(\vec{R_1}\times\vec{R_2}\right)\right] = A\left[\vec{r_1}\vdots\vec{r_2}\vdots\left(\vec{r_1}\times\vec{r_2}\right)\right]$$

where $\vdots$ have been used to separate different column vectors. If we now define,

$$\hat{S} = \frac{\vec{R_1}}{\left\|\vec{R_1}\right\|}$$

$$\hat{s} = \frac{\vec{r_1}}{\|\vec{r_1}\|}$$

$$\hat{M} = \frac{\vec{R_1} \times \vec{R_2}}{\left\|\vec{R_1} \times \vec{R_2}\right\|}$$

$$\hat{m} = \frac{\vec{r_1} \times \vec{r_2}}{\|\vec{r_1} \times \vec{r_2}\|}$$

we can find an estimate for the matrix A given by the equation,

$$\hat{A} = \left[\hat{S} \vdots \hat{M} \vdots \left(\hat{S} \times \hat{M}\right)\right] \left[\hat{s} \vdots \hat{m} \vdots (\hat{s} \times \hat{m})\right]^T$$

Except of the computational benefit of using this method, it is important to note that there is no propagation of floating point errors. If we instead accumulated the the total rotation of each particle, an error would be added at each rotation.

2.3.4 The Cell Linked List with Improved Time Coherency

Having to check for collisions every particle pair every time a particle is moved would take $O(n^2)$ time. This is not efficient and thus we need to use a method that takes advantage of the locality of the move and the sizes of the particles. One such method, which we used in our simulations, is called the cell linked list (CLL for short). The CLL is a form of spatial subdivision and it is named as it is because it uses a linked-list data structure. The simulation

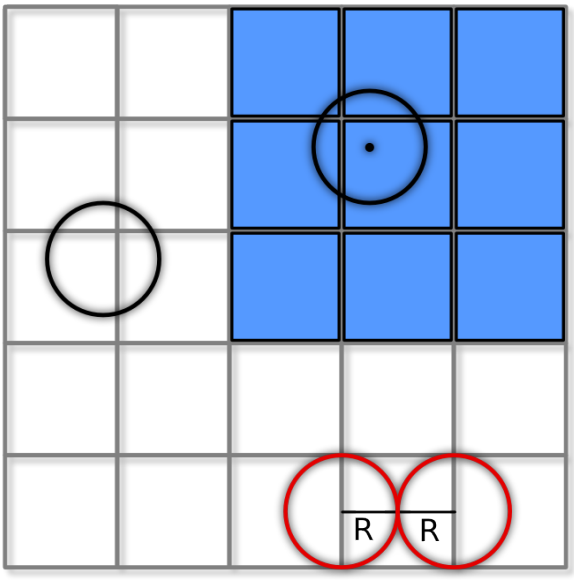


Figure 4: An example of a 2D grid. The red circles show that for a grid with cell size of $2R$, where R is the circumscribed radius, particles in next-neighboring cells can't touch each other. The blue squares show the cells that are checked for collisions for a particle with its centroid being the dot.

box is divided in a grid as seen in fig. 4 and we thus only need to check for collisions in a neighborhood of the particle. The neighborhood of a cell is defined as the 8 nearest neighbors

surrounding the cell for 2D, while in 3D there are 26 such neighbors. Particles are identified with only one cell using the position of their centroids. To facilitate this process we also use a minimum bounding sphere and a maximal inscribed sphere with a corresponding out-radius and in-radius. An example of an inscribed and a circumscribed sphere can be seen in fig. 5. When two particles have a distance smaller than 2 times the inscribed sphere radius, it is certain the two particles overlap and the move can be rejected. When the two particles have instead a distance smaller than 2 times the circumscribed radius, we have a possible collision which is resolved using the GJK algorithm, which is discussed in the next section. No overlap is possible in case the distance is greater than 2 times the circumscribed radius.

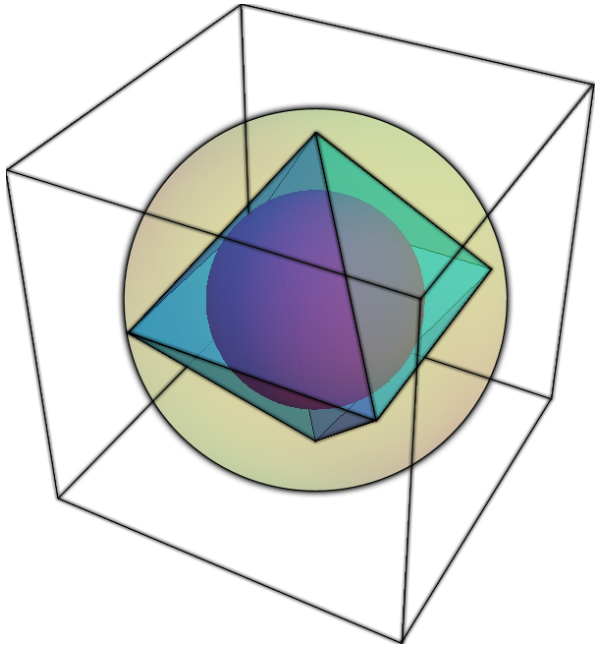


Figure 5: Inscribed (inner) and circumscribed (outer) sphere of an Octahedron

For the CLL, we choose a cell size of at least $2R$ plus some small offset (to avoid problems due to floating point errors), where R is the circumscribed radius of the particle so that we only have to search for collisions in the first neighboring cells of the cell the particle resides in since no particles in next-neighboring cells can touch as demonstrated with the red circles in fig. 4. In the case of the floppy box, one can derive formulas which tell one what size the cells should be for one particle to exactly fit inside. Each cell has a unique identifier and each particle resides in only one cell identified by the position of its centroid. Multiple particles may reside in the same cell without intersecting each other. Each cell also has 27 neighbors including itself in 3D (as a reminder, periodic boundary conditions are used) and these cells are searched sequentially when we check for collisions. The use of the linked-list data structure is trivial, each cell points to a particle inside it (if there is any) and each particle points to the next particle in the cell. A tail is used to identify the end of linked list.

The CLL reduces the complexity of searching for collisions to about $O(n \log n)$ but there is a further improvement that can be made which could not be found in the literature. This improvement takes advantage of the time coherency of the simulation, if a particle happened to overlap in a previous simulation step with a particle in a specific cell, it is more likely to collide with that particle or even some other particle in that cell. To take advantage of

this, we can keep a list of the neighboring cell ids for each cell. When a particle in cell A

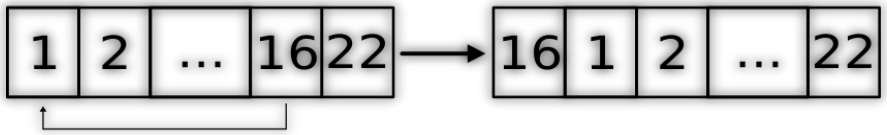


Figure 6: Example of pushing cell 16 to the front of the cell neighbor list

collides with particle in cell B, the B cell in the A's neighboring list gets pushed to the front (see fig. 6) of the list so that it can be checked first next time. Since the trial move gets rejected immediately after detecting a collision, this method provides shorter exit times. In our NVT simulations we observed about 20% performance increase. We used a simple array but a linked list data structure may also be used but this reduces the CPU cache coherency.

2.3.5 The Gilbert-Johnson-Keerthi Collision Detection Algorithm

In the last section we explained how we determine probable collisions using the cell linked list and the in- and circumscribed sphere radii. We have to further examine these probable collisions using a collision algorithm. In our simulations we have chosen the Gilbert-Johnson-Keerthi (GJK) algorithm which is one of the fastest and robust collision detection algorithms [36]. The GJK algorithm is an iterative method for computing the distance between convex objects. The attractiveness of GJK lies in its simplicity, which makes it fairly easy to implement, and its applicability to general convex polytopes.

This algorithm uses the fact that the shortest distance between two convex polygons is the shortest distance between their Minkowski difference and the origin. If the Minkowski difference encompasses the origin then the two objects intersect. The Minkowski sum is simply the sum of two sets of points (i.e. vertices), say A and B. That is, every point in set A is summed with every point in set B. More formally:

$$A \oplus B = \{\mathbf{a} + \mathbf{b} \mid \mathbf{a} \in A, \mathbf{b} \in B\}$$

The Minkowski difference is just the Minkowski sum where one of the sets of points is negative. The Minkowski difference is then the convex hull formed by these points.

$$A \ominus B = A \oplus (-B)$$

For collision detection the algorithm starts by picking a point in the Minkowski Difference. This point can be randomly chosen or chosen based on previous knowledge. It then iteratively builds a simplex trying to encompass the origin. If it succeeds it reports a collision, else it terminates and reports that no collision was detected. An example of a simplex that encompasses the origin in the GJK algorithm can be seen in fig. 7. The GJK algorithm also supports a caching of a separating axis which can be used if the simulation is time coherent. In that case, the algorithm will exit faster. We used this kind of caching for all particle pairs storing them using a hash table.

In pseudo-code, the GJK algorithm can be summed up in listing 1, where conv(X) means the convex hull of the set of points in X. In mathematics, the convex hull or convex envelope of a set X of points in the Euclidean plane or Euclidean space is the smallest convex set that contains X. The algorithm uses a support function $S_{A \ominus B}$ which finds the farthest/closest

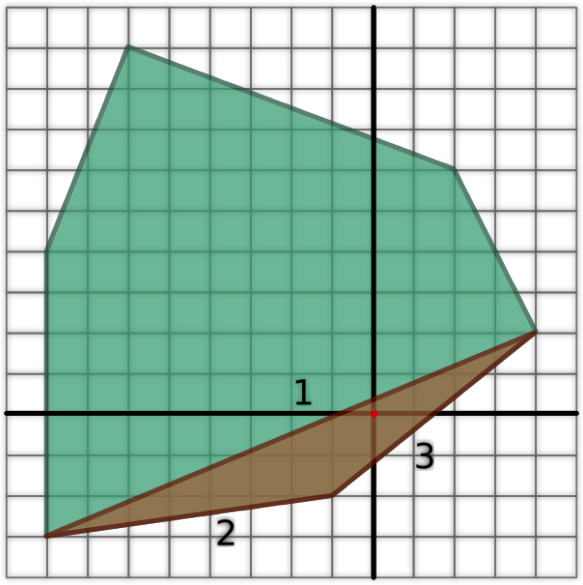


Figure 7: Example of a terminating simplex in the GJK algorithm in 2D. The green polytope is the Minkowski difference of two penetrating polytopes. The brown triangle is the simplex that encompasses the origin. Source [37]

Listing 1: The GJK algorithm pseudo-code

```
v := arbitrary point in A⊖B;
W := ø;
Y := ø;
repeat
    w := S_{A⊖B}(−v);
    if w ∈ Y or v·w > 0 then
        /* v is considered a separating axis */
        return false;
    Y := W ∪ {w};
    v := v(conv(Y));
    W := smallest X ⊆ Y such that v ∈ conv(X);
until |W| = 4 or ||v||^2 ≤ ε_tol max{||y||^2 : y ∈ W};
/* v is considered zero */
return true;
```

point of a polytope in a given direction. An important property of the support function is that,

$$S_{A\ominus B}(\mathbf{v}) = S_{A\oplus(-B)}(\mathbf{v}) = S_A(\mathbf{v}) + S_B(-\mathbf{v}).$$

Since we have to do with convex polytopes, we used a hill-climbing algorithm to further improve the performance.

In less mathematical terms, the algorithm can be described as follows. We start with an arbitrary point in $A \ominus B$ and find a point in the opposite direction using the support function. What this achieves is to try to pass through the origin. If the origin is past this new point, there is no intersection and we stop, else we add this point to our set Y and find a new direction to look for a support point. The goal of the algorithm is to build a simplex (i.e. a Tetrahedron in 3D) that contains the origin. If we find that it the origin is contained, a collision is detected. We advise the reader to check reference [38] which has a very thorough explanation of the algorithm.

Before implementing the GJK algorithm for use in our simulations, we first implemented the so called Separating Axis Theorem (SAT for short). SAT is simpler conceptually and thus easier to implement. The effectiveness of our GJK algorithm implementation was first tested against the SAT implementation and we found they had a 100% agreement in all test cases.

2.3.6 Center of Mass

Due to finite-size effects, the center of mass (CM) of the ensemble can drift away from its initial position. This is an unwanted situation since it introduces an unphysical collective motion of the particle ensemble. To counter such a situation, we keep track of the center of mass and extract it from the particle positions. To do this, the center of mass is calculated at the start of the simulation using,

$$\vec{R}_{cm} = \sum_i \vec{r}_i.$$

We then keep track of the "real" coordinates of the particles, i.e. the coordinates without the periodic boundary conditions applied. When a particle is displaced, by $d\vec{x}$, $\vec{R}_{cm}$ is displaced by $d\vec{x}/N$, where N is the number of particles. Lastly, when we need to print the particle coordinates, we remove the center of mass $\vec{R}_{cm}$ from the "real" coordinates of each particle and then apply the periodic boundary conditions.

We should note that from our simulations of hard polyhedra, we found the displacement of the center of mass to be less than $10^{-15}L$, where L is the length of the box for up to 10^7 MC steps.

2.4 Correlation Functions, Fragility and Crystallinity

Correlation functions are useful statistical quantities which measure, as their name suggests, the correlation between two or more random variables in either space or time. Correlation functions are a useful indicator of how the dependency of the correlated variables, changes over space or time. In this thesis, we have used various such correlation functions which were useful in the investigation of the various properties of the particles. We present these functions and other concepts here for the convenience of the reader. It should be noted that these quantities do not take into account the orientational degrees of freedom of the particles, which are here assumed to be point-like.

2.4.1 Radial Distribution Function

We start by introducing the radial distribution function (or pair correlation function), which is the basis for most of the other correlation functions we have used in the analysis of our data. The radial distribution function, describes how density varies as a function of distance from a reference particle. Given a homogeneous system of N particles, the function itself is defined as,

$$g(r) = \frac{1}{\rho^2}\left\langle \sum_{i=1}^{N}\sum_{j\neq i}^{N} \delta(\mathbf{r}_i)\delta(\mathbf{r}_j - \mathbf{r}) \right\rangle = \frac{V}{N^2}\left\langle \sum_{i=1}^{N}\sum_{j\neq i}^{N} \delta(\mathbf{r} - \mathbf{r}_{ij}) \right\rangle, \tag{3}$$

where $r_{ij} = r_i - r_j$ and $\langle . \rangle$ denotes an ensemble average. As we described in the introduction, the glass phase is characterized by the particles forming "cages" from which particles take a long time to escape. The cages are identified with particles clustering around a particle at a given radius. Thus the radial distribution function is the perfect measure for this. An example of a radial distribution function plotted with respect to the reduce radius r/σ , where σ is the double radius of a spherical particle, can be seen below. The peaks in the plot, show

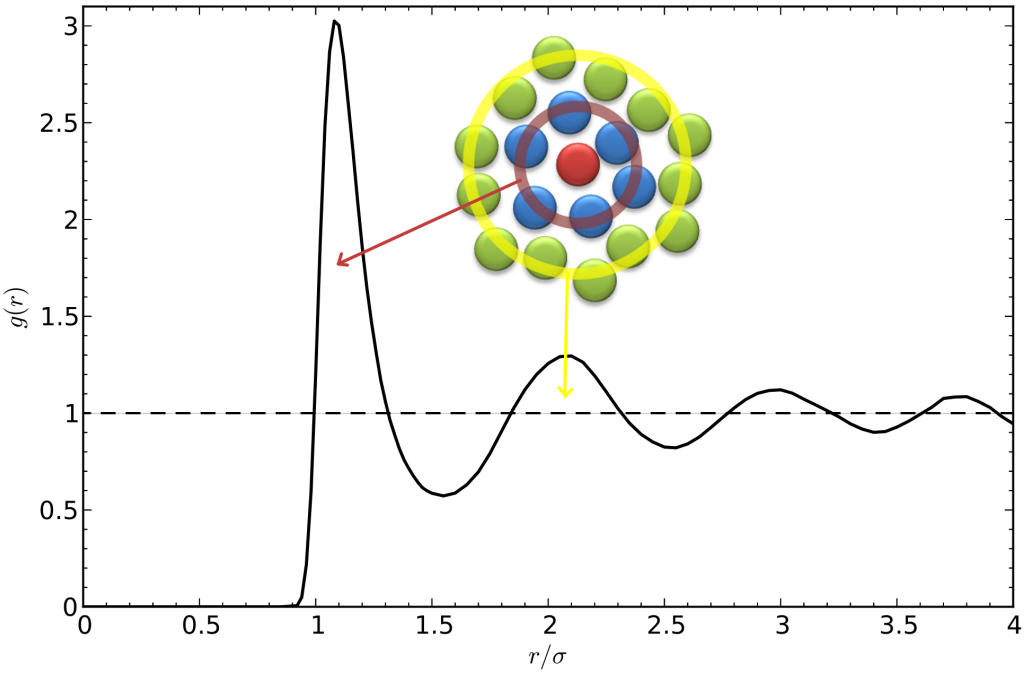


Figure 8: Radial distribution function for the Lennard-Jones model fluid at $T^* = 0.71$, $n^* = 0.844$. In the inset we have drawn a sketch showing how the peaks may correspond to rings of higher density at a given distance from a reference particle (depicted in red). Source (Wikipedia)

increased density e.g. at an average distance of 1.2 of each particle, we have the highest probability of finding a particle.

2.4.2 Static Structure Factor

Next we introduce the structure factor, or more specifically the static structure factor. In principle it is a mathematical description of how a material scatters incident radiation. It is essentially the Fourier transform of the radial distribution function and is defined as follows,

$$S(\mathbf{k}) = \frac{1}{N} \langle \rho_{\mathbf{k}} \rho_{-\mathbf{k}} \rangle = \frac{1}{N} \left\langle \sum_{lm} e^{-i\mathbf{k}(\mathbf{r}_l - \mathbf{r}_m)} \right\rangle, \tag{4}$$

where ρ_k is the Fourier transform of the microscopic density $\rho(\mathbf{r})$. For an isotropic fluid, the static structure factor, can be described with respect to the radial distribution function as,

$$S(\mathbf{k}) = 1 + \rho \int \exp(-i\mathbf{k}\mathbf{r}) g(r) d\mathbf{r}. \tag{5}$$

We can thus calculate the static structure factor either directly using eq. (4), or through the radial distribution function eq. (5). Calculating this quantity is similar to performing a scattering experiment. Thus, when we calculate it using eq. (4), we average over a shell at $|k|$ of thickness dk. Additionally, in our simulations, since the simulation box is of finite size, we have to consider only wave vector multiples of $\frac{2\pi}{L}$, where L is the length of the simulation box. Multiples of $\frac{2\pi}{L}$ are used because L is the largest possible length making $\frac{2\pi}{L}$ the smallest possible wave vector.

2.4.3 Self-Intermediate Scattering Function

The central quantity we measured in this thesis is the self part of the intermediate scattering function. The intermediate scattering function is the spatial Fourier transform of the *van Hove* function, which measures the probability density of finding a particle i in the vicinity of $\mathbf{r}$ at time t knowing that the particle was in the vicinity at the origin at time $t = 0$. The definition of the self-intermediate scattering function is very similar to that of the static structure factor,

$$F_s(\mathbf{k}, t) = \frac{1}{N} \left\langle \sum_j e^{-i\mathbf{k}(\mathbf{r}_j(t) - \mathbf{r}_j(t_0))} \right\rangle. \tag{6}$$

The first peak in the static structure factor is associated with the freezing or glass transition of a supercooled liquid [39]. Thus, in our analysis of the data, we measure the self-intermediate scattering function at $|k| = k_0$, the first peak in the static structure factor which gives us information on the structural relaxation of the particles. In fig. 9 an example of the Self-Intermediate Scattering Function (SISF for short) can be seen for the case of a simple liquid and for the case of a supercooled liquid. In the simple liquid case, a regular exponential decay is observed, while for the supercooled liquid, we can identify various regimes. More specifically, two decays are observed and the relaxation IIa is called the β relaxation and III the α relaxation. The two regimes are separated by a plateau, II as seen in the figure. The β relaxation is associated with a regime where particles appear trapped in cages. The decay from the plateau IIa may be fitted as $f + At^{-a}$ and the decay from the plateau IIb as $f - Bt^b$. The α relaxation regime is associated with the particles escaping the cages and the decay may by fitted with a stretched exponential $\sim e^{-(t/\tau)^\beta}$, where τ is the relaxation time and β is the stretching exponent. As can be deduced, it is this regime and the stretching

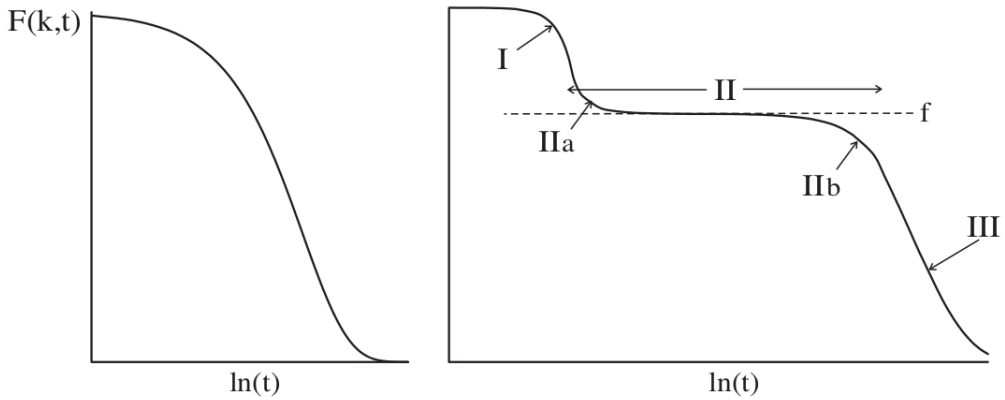


Figure 9: Left: $F_s(k,t)$ exhibiting exponential $e^{-t/\tau}$ decay for a normal liquid. Right: supercooled liquids do not have such a simple decay. The various temporal regimes are described in the text. Notice the logarithmic scale. Source [40]

behavior together with the plateau, that are characteristic of glassy behavior. An other school of thought is to characterize the glass regime as the regime where the α decay is not more experimentally accessible in a sensible time scale.

2.4.4 Fragility

Fragility characterizes how rapidly the dynamics of a material slow down as it is cooled toward the glass transition temperature. Materials with higher fragility have a relatively narrow glass transition temperature range, while those with low fragility have a relatively broad glass transition temperature range. More specifically, one can use the relaxation time described above. Fragile liquids then are highly sensitive to temperature change, while the opposite is true for strong ones [41]. Physically, fragility may be related to the presence of dynamic heterogeneity in glasses, as well as to the breakdown of the usual Stokes-Einstein relationship between viscosity and diffusion. The concept of fragility is best summarized in a renormalized Arrhenius plot, where the temperature is rescaled by the glass transition temperature T_g and fragility is defined by the logarithmic slope at T_g [41]. An Arrhenius plot displays the logarithm of kinetic constants plotted against the inverse temperature . In colloidal suspensions, packing fraction or concentration can be used equivalently to temperature. The data plotted as described, can be fitted with an Arrhenius equation, or with the Vogel Fulcher Tammann (VFT) equation. The VFT equation, is given by,

$$\tau_a = \tau_0 \exp\left(\frac{A}{\phi_0 - \phi}\right), \tag{7}$$

while the Arrhenius equation is given by,

$$\tau_a = \tau_0 \exp\left(C\phi\right), \tag{8}$$

where ϕ is the packing fraction (concentration). Using these tools, we will be able to tell if hard polyhedra are strong or fragile glass formers and how this changes for different polyhedra.

2.4.5 Crystallinity

Some of the particle shapes we studied, have a chance to crystallize. Because we were interested in the glass phase which is amorphous, we needed some way of being able to quickly tell if a sample has crystallized or not. We used the so called Bond Orientational Order Parameter [42]. Two distinct broken symmetries distinguish crystalline solids from isotropic liquids. Broken translational invariance is measured by the phase of the periodic density modulations in a solid. A broken rotational symmetry is defined by the singled-out crystallographic axes. These two symmetries are not independent, because rotating one patch of perfect crystal relative to another clearly disrupts not only orientational correlations, but translational correlations as well. A relative translation of the two patches, on the other hand, decorrelates translational order, but leave orientational correlations intact.

The definition of a bond is introduced to define the bond orientational order parameter. Bonds of course do not refer to the traditional chemical bonds, but rather lines connecting near-neighbors, where the definition of "near" can be specified in various ways. With this definition of a bond, we consider the average bond-orientational order parameter for particle i,

$$q_{lm}(i) = \frac{1}{N_b(i)} \sum_{j=1}^{N_b(i)} Y_{lm}(\theta_{ij}, \phi_{ij}), \tag{9}$$

where $N_b(i)$ is the number of the particle's near-neighbors and Y_{lm} are the spherical harmonics. θ_{ij} and ϕ_{ij} are the polar and azimuthal angles of the bond formed by particles i and j as explained before. The spherical harmonics for a given value l (and $|m| \leq l$) form a $(2l+1)$-dimensional representation of the rotational group $SO(3)$. This means that the $q_{lm}(i)$ corresponding to a particular representation are scrambled by rotating the external coordinate system.

Solid-like particles are identified as particles for which the number of bonds per particle $\xi(i)$ is at least ξ_c, where

$$\xi(i) = \sum_{j=1}^{N_b(i)} H[d_l(i,j) - d_c], \tag{10}$$

where H is the Heaviside function, d_c is the dot product cutoff and,

$$d_l(i,j) = \frac{\sum_{m=-l}^{l} q_{lm}(i) q_{lm}^*(j)}{\left(\sum_{m=-l}^{l} |q_{lm}(i)|^2 \right)^{1/2} \left(\sum_{m=-l}^{l} |q_{lm}(j)|^2 \right)^{1/2}}, \tag{11}$$

which is normalized to 1. Crystallinity can be then defined as the number fraction of solid-like particles.

There are a few parameters that need to be tweaked. For example, the cutoff distance can be defined as $1.2r_0$, with r_0 the position of the first peak in the radial distribution function. The parameter l can be set depending on the order we expect. For example for hexatic order, we set $l = 6$ and so on. As for the rest of the parameters, i.e. the dot-product cutoff and ξ_c can be tweaked by using the fact that in the liquid-solid coexistence packing fraction (concentration), crystallinity should be about 0.5.

2.5 Mode Coupling Theory

We will give a brief introduction the the mode coupling theory of the glass transition. We will not go too much into detail but instead try to focus on the predictions the theory makes that are relevant to our research project.

In the seventies, a considerable theoretical effort was undertaken in order to find a correct quantitative description of the dynamics of dense simple liquids. By using mode-coupling approximations [43], equations of motion for density correlation functions, described in more detail below, were derived and it was shown that their solutions give at least a semi-quantitative description of the dynamics of simple liquids in the vicinity of the triple point. In particular it was shown that these equations give a qualitatively correct description of the so-called cage effect, i.e. the phenomenon that in a dense liquid a tagged particle is temporarily trapped by its neighbors and that it takes the particle some time to escape this cage. A few years later Bengtzelius, Götze and Sjölander (BGS) simplified these equations by neglecting some terms which they argued were irrelevant at low temperatures [39]. They showed that the time dependence of the solution of these simplified equations changes discontinuously if the temperature falls below a critical value Tc . Since this discontinuity was accompanied by a diverging relaxation time of the time correlation functions, this singularity was tentatively identified with the glass transition.

The dynamics of liquids are usually described by means of the intermediate scattering function $F(\mathbf{k},t)$ as,

$$F(\mathbf{k},t) = \frac{1}{N}\left\langle \delta\rho^*(\mathbf{k},t)\delta\rho(\mathbf{k},0)\right\rangle \quad with \quad \rho(\mathbf{k},t) = \sum_{j=1}^{N} e^{i\mathbf{k}\cdot\mathbf{r_j}(t)}$$

where N is the number of particles and $\mathbf{r}_j$ is the position of particle j at time t. For an isotropic system, the equations of motion for the correlator mentioned above can be written as

$$\ddot{F}(k,t) + \Omega^2(k)F(k,t) + \int_0^t dt' \left[M^0(k,t-t') + \Omega^2(k)m(k,t-t')\right]\dot{F}(k,t') = 0. \tag{12}$$

Ω is a microscopic frequency which can be computed from the structure factor $S(k)$ via $\Omega^2(k) = k^2k_BT/mS(k)$. The kernel $M^0(k,t)$ describes the dynamics at short times and gives the only relevant contribution to the integral at temperatures in the vicinity of the triple point, while the kernel $m(k,t)$ becomes relevant at temperatures where the system is strongly supercooled. It is thus important to note here that, the short time characteristics of the system, don't affect the long time evolution of the correlator $F(k,t)$. If we further assume that the kernel $M^0(k,t)$ is sharply peaked at $t = 0$ and thus can be approximated by a δ-function, $M^0(k,t) \approx \nu(k)\delta(t)$, we recognize eq. (12) as a damped harmonic oscillator with the addition of of a retarded friction proportional to $m(k,t)$. These equations are exact, since the aforementioned kernels have not been specified yet. In the approximations of the *idealized* MCT, the kernel $m(k,t)$ is expressed as a quadratic form of the correlator, $\sum_{\mathbf{q}+\mathbf{p}=\mathbf{k}} V(\mathbf{k};\mathbf{q},\mathbf{p})F(\mathbf{q},t)F(\mathbf{p},t)$, where the vertices $V(\mathbf{k};\mathbf{q},\mathbf{p})$ can be computed from the structure factor. With this approximation one therefore arrives at a closed set of coupled equations for $F(\mathbf{k},t)$, the so-called mode coupling equations, whose solutions thus give the full time dependence of the intermediate scattering function. These are the equations proposed by BGS [39] and it is believed they give a self-consistent description of the dynamics of a particle.

We will not go into more detail of the specifics of the theory but we'll rather present the important predictions MCT makes. It should be noted that MCT has to make many approximations to get analytical solutions and thus is not fit for quantitative predictions but makes a set of very important qualitative predictions.

One of the main predictions of the idealized version of MCT is that there exists a critical temperature T_c at which the self diffusion constant D, or the inverse of the α-relaxation times τ of the correlation functions, vanishes with a power-law,

$$D \propto \tau^{-1} \propto (T - T_c)^{\gamma}, \tag{13}$$

where the exponent γ is larger than 1.5.

Apart from the existence of a critical temperature T_c , one of the most important predictions of the theory is the existence of three different relaxation processes, which we have already seen in fig. 9. The first one is just the trivial relaxation on the microscopic time scale. Since on this time scale the microscopic details of the system are very important, hardly any general predictions can be made for this time regime. This is different for the second and third relaxation processes, the aforementioned β and α-relaxation. For these time regimes MCT makes quite a few predictions. For the α-relaxation we're mostly interested, MCT predicts a result called the *time-temperature superposition principle.*This means that the correlators for different temperatures can be collapsed onto a master curve if they are plotted versus t/τ , where τ is the α-relaxation time. An example of this can be seen in fig. 10. Also, although

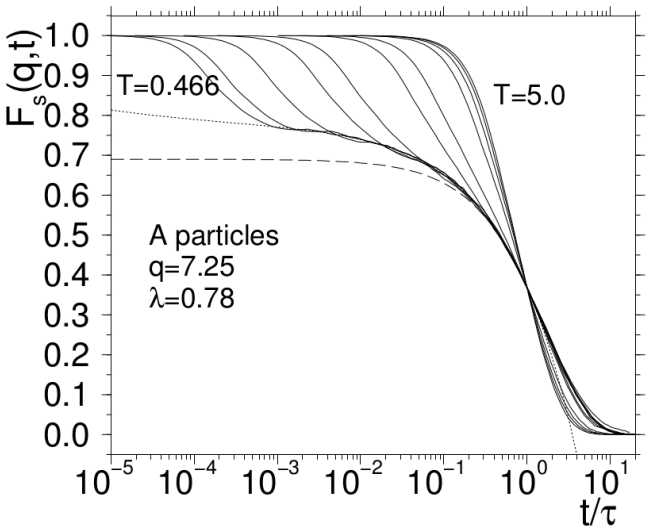


Figure 10: SISFs vs t/τ, where τ is the α-relaxation time. The curves are collapsed onto a master curve indicated by the dotted line. Source: [44]

MCT predicts that the shape of the relaxation curves will depend on the correlation function considered, the theory also predicts that all of them share a common property, namely that all the corresponding relaxation times diverge at T_c with a power-law whose exponent γ is independent of the correlator, see eq. (13). Also this prediction was confirmed in experiments and in computer simulations [45, 46].

It is interesting to note that most of the systems for which MCT gives a good description of the dynamics belonging to the class of fragile glass formers i.e. systems that show a significant

change of their activation energy if their viscosity is plotted in an Arrhenius plot. From the point of view of MCT the dynamics of supercooled liquids can be described as follows: In the temperature range where the liquid is already supercooled, but which is still above T_c , the dynamics is described well by the idealized version of MCT. In the vicinity of T_c an extended version of the theory has to be used, whereas for temperatures much below T_c it is likely that the simple way MCT takes into account the hopping processes is no longer adequate and a more sophisticated theory has to be used. With hopping we refer to the process in which an individual particle has enough energy to overcome its cage although the average energy of the system may not be sufficient.

2.6 Time in Monte Carlo Simulations

Since its introduction, the Metropolis algorithm has had a great success because of its ability to introduce unphysical moves, provided they obey detailed balance. Unphysical moves such as cluster moves or biased sampling [34] greatly speed up the equilibration process, making MC simulations a powerful tool for the study of equilibrium properties.

On the other hand molecular dynamics (MD) and Brownian dynamics (BD) are usually the techniques of choice to the study dynamical properties of systems such as molecular and colloidal systems. Unfortunately, MD and BD have some problems incorporating non-differentiable potentials (i.e. hard potentials) and usually a different approach, such as the event driven molecular dynamics simulation is used [47]. Accounting for rotations is even more complex and many times a stochastic approach is used instead of integrating the equations of motion. To circumvent these limitations, there has been a continuous effort to find how the "time" evolution in Monte Carlo Simulations can be mapped to that of BD and MD [48–51]. The problem has had many different approaches, but the general consensus is that a MC scheme that is desired to be mapped to MD/BD, should not contain any unphysical moves. It should be noted though, that although MD and BD provide a more appropriate description of dynamics, there is still difference in time in MD and BD units compared to the physical time measured in experiment for some dynamic property. This discrepancy is usually just a factor but the general dynamics remain the same. Thus, finding a simple scaling of time between MD/BD and MC should be sufficient.

Work done by Sanz et al. [50], focuses on the fact that a "link" between the different dynamics can be created using the Einstein diffusion relation [52], which relates the mean square displacement in three dimensions to the diffusion coefficient and time as follows:

$$\left\langle r^2 \right\rangle = \lim_{t\to\infty} 6Dt$$

We shall not go into detail on how this connection has been done but a very good agreement has been reported using this approach and using an acceptance of more than 70% for translation moves.

On the other hand, Kob et al. [51] has reported a very good agreement up to a scaling factor of dynamic properties at the long time scale. Kob confirmed the predictions made by MCT which we mentioned in the last subsection, namely that long time average dynamics are not influenced by the short time properties of the system. Thus all simulation methods like MC/MD/BD etc. should agree up to some factor in the long time range of the system. In the scheme Kob et al. employed in their paper, they choose a maximum displacement corresponding to the maximum displacement at which the relaxation time for a dense system

shows a minimum. In fig. 11 we see that there is very good agreement of MC with other

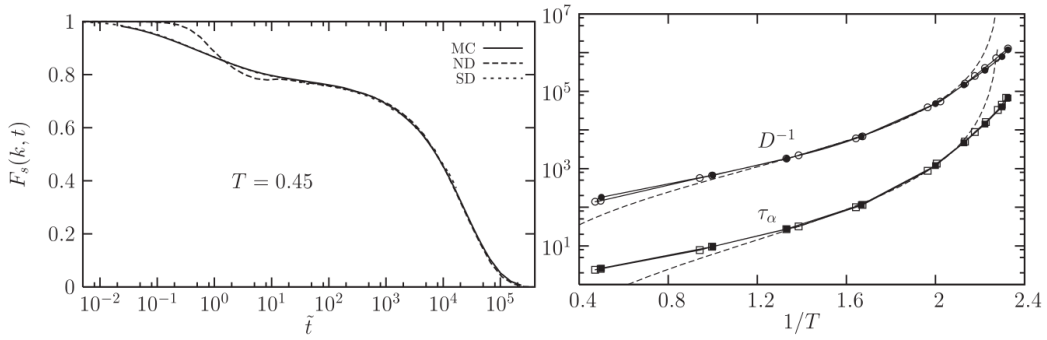


Figure 11: Left: comparison of the self-intermediate scattering function for k = 7.21, and T = 0.45, obtained in Monte Carlo (MC) dynamics, Newtonian dynamics (ND), and stochastic dynamics (SD). Time is rescaled to obtain maximum overlap at large times. MC and SD agree over the complete time range (indeed the SD dotted line is barely visible below the full MC line), while MC and ND only agree when $F_s(k,t)$ is close to the plateau and below. The dip observed at short time in ND is due to thermal vibrations suppressed in both SD and MC. Right: temperature evolution of the alpha-relaxation time τ_α (T) and the inverse of the self-diffusion constant $1/D(T)$ in an Arrhenius plot. Open symbols are for ND, closed symbols for MC (vertically shifted to obtain maximum overlap with ND data), and the dashed lines are power-law fits to a divergence at $T_c = 0.435$. Source [51].

stochastic simulations in the long time range. In the results section that follows, we chose to use this scheme of choosing a maximum displacement because we're only interested in the long time behavior of our particle systems. As it shall become obvious later on, we don't mind an overall scaling factor, as long as the relative behavior stays close to reality. It should also be noted that the SISF of the Monte Carlo simulations don't exhibit a very pronounced plateau as opposed to the other schemes.

Generally, the aforementioned effort of the community has been mainly focused on particles such as spheres which are highly symmetrical and do not possess orientational degrees of freedom. The story may be different for particle shapes such as polyhedra, investigated here, but unfortunately, because of the limited time available, pursuing this investigation is not possible. It is though useful to state some theoretical considerations. In the limit where the maximum displacement is chosen infinitely small, the diffusion coefficient of the solute under the MC scheme, should be

$$D_0 = \frac{\delta l^2}{6}, \tag{14}$$

where δl is the maximum displacement. In reality, we choose a finite maximum displacement which means that a displacement move has a chance to be rejected due to overlaps. Thus, the mean squared displacement should be calculated as,

$$\left\langle \vec{r}^2 \right\rangle = \frac{\int_V d\vec{r} a(\vec{r}) \vec{r}^2}{\int_V d\vec{r} a(\vec{r})} = \frac{8\delta l^3}{\bar{a}} \int_V d\vec{r} a(\vec{r}) \vec{r}^2, \tag{15}$$

where $a(\vec{r})$ is the acceptance and the volume V is defined as $(2\delta l)^3$. We also have intuitively that, $\lim_{\delta l \to 0} a(\vec{r}) = \bar{a} = 1$, but for finite maximum displacements the acceptance has a non-trivial dependence on $\vec{r}$. In the results section we will investigate briefly the behavior of the acceptance, but it would be interesting to derive a dependence at least with orientational

degrees of freedom integrated out, which could be used to scale time for any maximum displacements.

3

Results

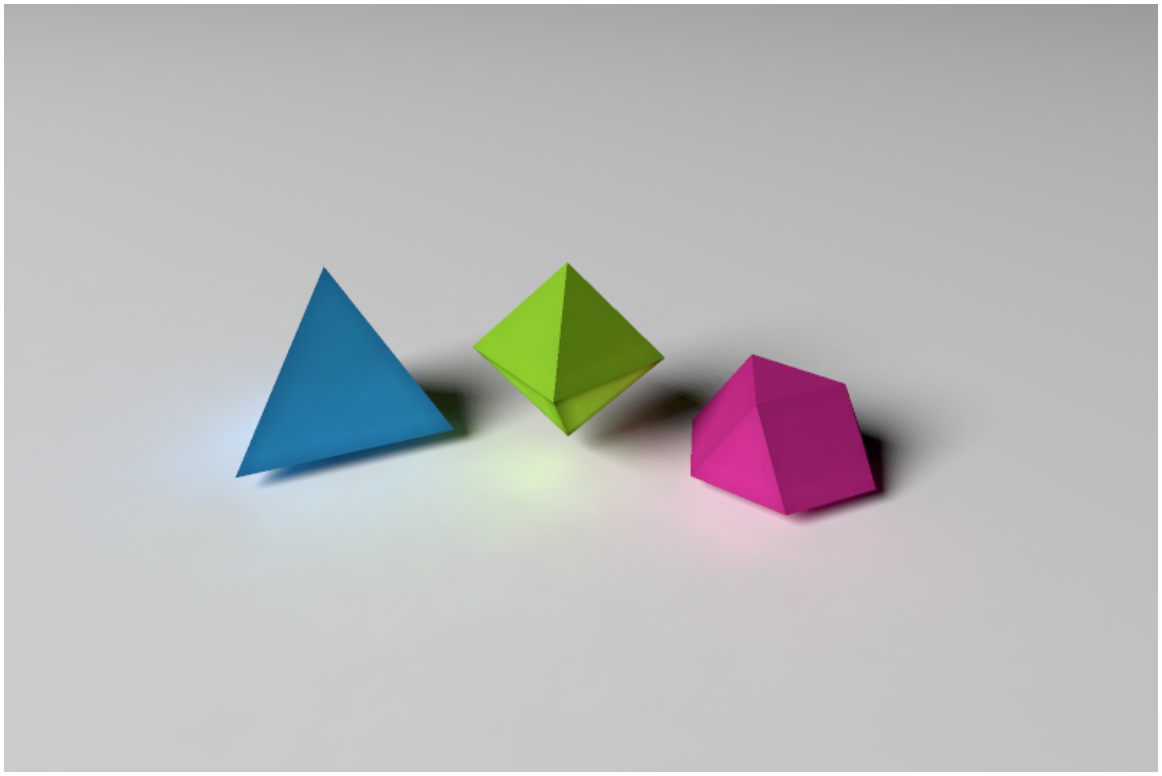


Figure 12: The three particle shapes studied in this thesis. From left to right: Tetrahedron, Octahedron, Triangular Cupola.

In this thesis, we have studied three different polyhedra, namely Octahedra, Tetrahedra and Triangular Cupola as seen in fig. 12. The first two are Platonic Solids and the Triangular Cupola is a Johnson solid. Octahedra exhibit point symmetry while Triangular Cupola are the most asymmetric and can be seen as one half of a Cuboctahedron, which is an Archimidean solid. We chose Octahedra and Tetrahedra because they can be synthesized and studied experimentally and Triangular Cupola to have a comparison to a particle shape that is known [26] to remain in the an amorphous state. In all of our experiments, unless specified, polyhedra of unit volume have been used, and some of their geometric properties are summed up in table 1.

	Polyhedra Characteristics				
$Name$	R_{in}	R_{circ}	N_V	N_E	N_F
Octahedron	0.524557532	0.908560296	6	12	8
Tetrahedron	0.416341589	1.249024766	4	6	4
Triangular Cupola	0.0	0.946721157	9	15	8

Table 1: Geometric characteristics of the three polyhedra studied in this work. From left to right, we have the in-radius R_{in}, the circum-radius R_{circ}, the number of vertices N_V, the number of edges N_E and the number of faces N_F for the named Polyhedra, of unit volume.

We wish to note here that the Triangular Cupola has a zero in-radius as its center was defined on its hexagonal face. It could be defined inside its volume, but it would increase the circum-radius, which was considered to be less computationally efficient.

3.1 Equations of State and General Properties

We started by running NPT simulations to extract the equations of state. The equations of state were extracted to determine the pressure ranges for each particle and to compare them with available data. The simulations were performed using 2000 particles. A compression run was performed by starting from a sample of low packing fraction of 0.1. Because the starting sample was not random, a small NVT run of about 10^4 steps was used before proceeding with the NPT simulation. In the NPT simulation, pressure was varied by some constant and subsequently equilibrated for about 10^6 steps. We used acceptances of 30% for translations and rotations and 15% for volume moves. About $6 \cdot 10^5$ of the last MC steps at each pressure point were used for calculating the average packing fraction.

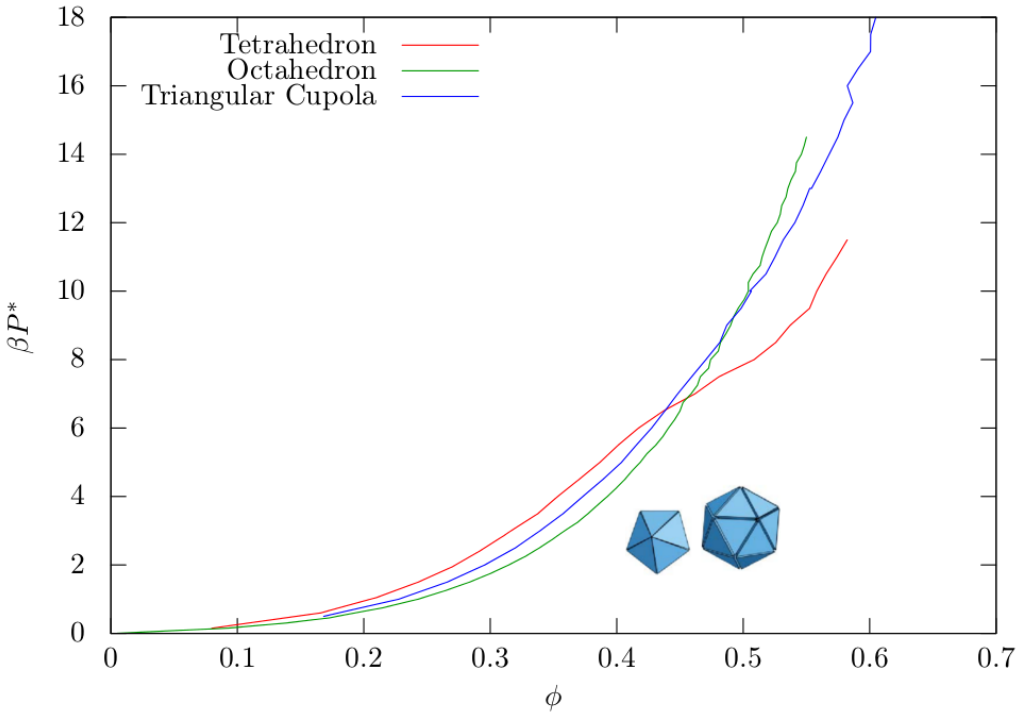


Figure 13: Equations of state for the three particles studied in this Thesis. Equations of State were acquired by NPT Monte Carlo simulations using compression. Tetrahedra exhibit a small plateau at a packing fraction of about 0.45.

In fig. 13 we show the equations of state for the three particle shapes studied in this thesis. We plot the reduced pressure, $\beta P^* = \beta P V_p$ as a function of packing fraction, with V_p the volume of the particle. Tetrahedra exhibit a small plateau at a packing fraction $\phi \approx 0.45$.

From [29], we know this is a phase transition to either a more complex liquid or a quasi-crystal, depending on the compression rate. Triangular Cupola is the only one of the three particle shapes that did not exhibit a phase transition in the time frame of our simulations, although two Triangular Cupola can form together a Cuboctahedron, which is known to crystallize into a face-centered cubic-like and a body-centered cubic phase [26]. Octahedra on the other hand, are known to exhibit two phases, a metastable BCC phase and a Minkowski Crystal [28, 53]. Typical configurations of the isotropic phases of the particles as well as the Minkowski crystal of Octahedra, can be seen in fig. 14. At high densities, Tetrahedra form small clusters of decahedra and pentagonal-dipiramids. The formation of these clusters, leads to the jump in packing fraction seen in fig. 13. Triangular Cupola often align their hexagonal faces forming dimers.

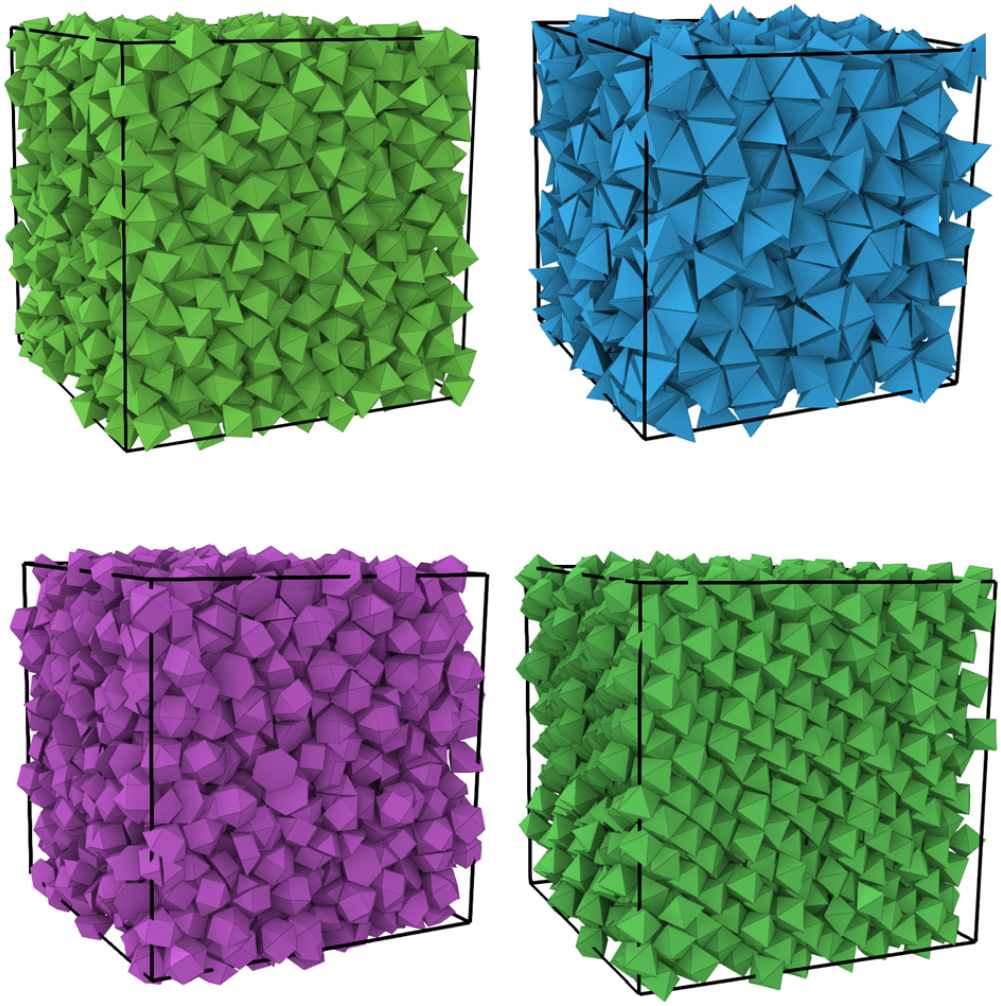


Figure 14: Typical configurations of the isotropic phases of the three particles. **Lower Right corner**: A Minkowski lattice of Octahedra from one of our simulations.

3.2 Radial Distribution Functions and Static Structure Factors

To prepare samples for extracting the radial distribution functions and the static structure factors, a quenching process was introduced, in which a sample at packing fraction $\phi \approx 0.1$ is gradually brought to the desired packing fraction by rapidly changing the pressure in each MC step. This process is repeated many times and the resulting samples with the lowest crystallinity fraction, eq. (11), are chosen. We would like to mention here that although we used this quenching process, in test runs where we varied the pressure rate of change, no correlations between crystallinity and the rate of change of the pressure were found. This was checked for pressure rates of $\beta P^*/MCstep$ between 0.0001 and 0.0256. Slower rates were not examined since any slower rates would be too computationally expensive.

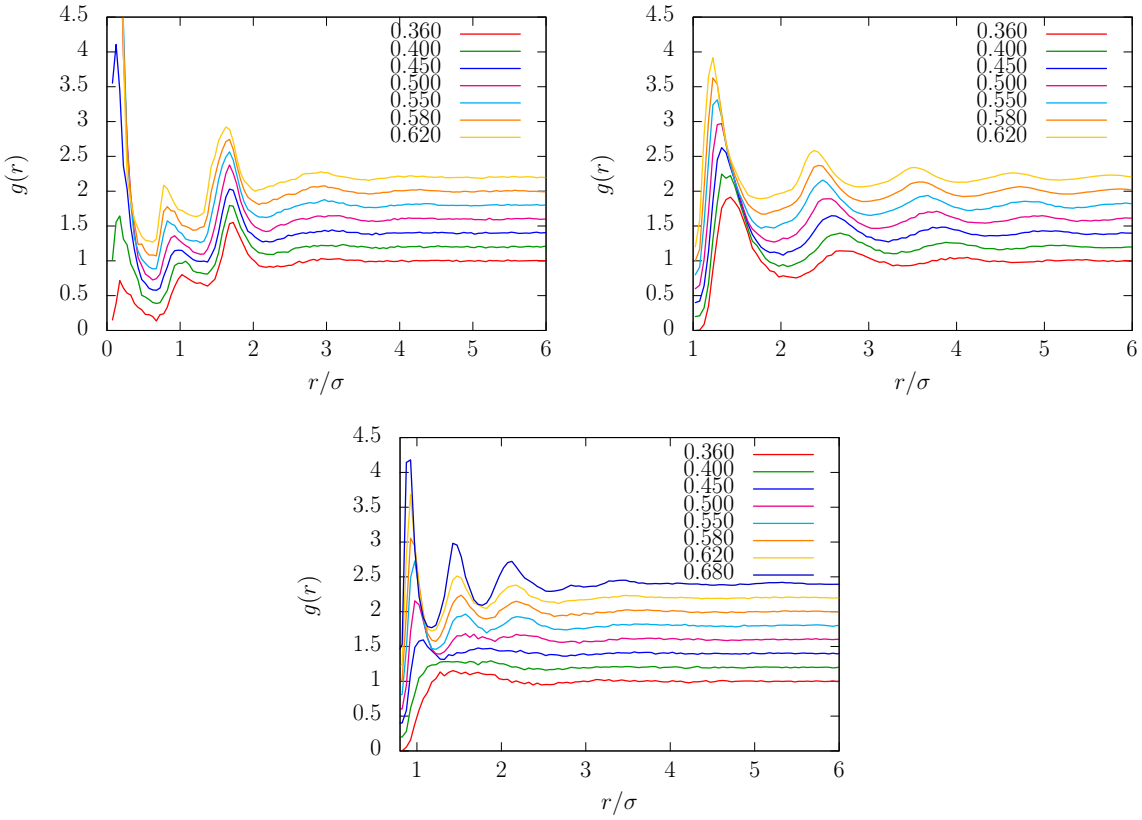


Figure 15: Radial Distribution Functions for various packing fractions listed inside the figure. The distribution functions have been shifted by 0.2 with respect to each other to make them distinguishable. **Top**: Triangular Cupola, Octahedra, **Bottom**: Tetrahedra

In fig. 15, we have plotted the radial distribution functions for the three particle shapes at different packing fractions. For all particles, it is obvious that the peaks get contracted to lower radii and become more pronounced as packing fraction increases. More particularly, Octahedra don't exhibit any extraordinary behavior, but for Tetrahedra and Triangular Cupola, there are some additional observations that can be made. Tetrahedra show a splitting of the second peak and the appearance of the first one, as packing fraction increases, which is probably connected to the formation of decahedra and pentagonal-dipiramids as we will discuss in the next section. Triangular Cupola show a very peculiar behavior owning to their

asymmetric shape. The peak corresponding to distances close to zero increases significantly with increasing packing fraction, as the number of particles with their hexagonal faces aligned increases. At $r \approx 1.0\sigma$, a 2nd peak is observed, which probably comes from particles aligning their hexagonal face with the opposite face of other particles (head to tail).

Static Structure Factors don't change much with packing fraction and since we're solely interested in the position of their first peak, only these peak positions are presented here. The peak positions will be used later to calculate the Self-Intermediate Scattering functions. The values we found are $k\sigma = 5.6$, 3.83 and 4.9 for Octahedra, Tetrahedra and Triangular Cupola respectively, with $\sigma = V_p^{1/3}$.

3.3 Relaxation and Short Time Diffusion Dependence on the Maximum Displacement

As was explained in Section 2, an appropriate maximum displacement had to be chosen for our simulations in order to relate the number of MC cycles with a physical time.

To get a better understanding of how the shape of the particles affects the acceptance probability of translational moves, we run a simulation of two particles which are initialized at positions sufficiently far apart from each other and with random orientations. They are then brought as close as possible (in 1 dimension) until they touch and their distance is measured. The distribution of these *minimum* distances d_{min} shows how much the particle shape affects the distribution of acceptance in some range, where the orientational degrees of freedom have been integrated out. In fig. 16 one can see the profiles that were extracted for 4 polyhedra, namely, Octahedra, Tetrahedra, Triangular Cupola and Cubes. The corresponding plot for Spheres, would be a delta function centered around $2R$. As can be seen, point symmetric particles like Cubes and Octahedra, have sharper peaks, Tetrahedra are more spread, while Triangular Cupola which is half of a Cuboctahedron, seems to have one sharp and one less sharp peak which is related to how the particle faces are aligned. This demonstrates how the acceptance ratio for a displacement move depends on radius and that in no way can we assume that it is constant and equal to the average acceptance. It also shows how we cannot completely avoid unphysical moves however small we choose to make the maximum displacement.

Next, the dependence of the relaxation time of the α-relaxation on the maximum displacement was studied. To this end, we calculated the Self-Intermediate Scattering Function (eq. (6)) and we fit the SISF with a stretched exponential $\sim e^{-(t/\tau)^\beta}$, as described in Section 2. To extract the SISFs, we ran NVT simulations at packing fractions $\phi = 0.55$ and 0.62 for Octahedra and 0.62 for the other two particle shapes. Runs were made with $1.01 \cdot 10^7$ MC steps for $\phi = 0.62$ and $1.1 \cdot 10^6$ MC steps for the lower packing fraction. For each maximum displacement, rotational acceptance was kept fixed at 30%. Snapshots of the configurations were taken at logarithmic time intervals. To get sufficiently smooth results, we averaged over approximately 5 runs. Since Octahedra had a chance to crystallize, more than 20 runs were needed to get samples that did not crystallize at sufficiently long relaxation times. We additionally averaged over 10 different starting times t_0 as seen in eq. (6). Samples were prepared using the quenching procedure described in the last sub-section. An example of the SISFs extracted using the above described method, can be seen in fig. 17. The maximum displacements are normalized by the box side length (cubic box) since we use reduced coordinates in our simulations. The box side for 0.62 packing fraction is 14.775. Their tails were fit with a stretched exponential to extract the relaxation time for each maximum displacement.

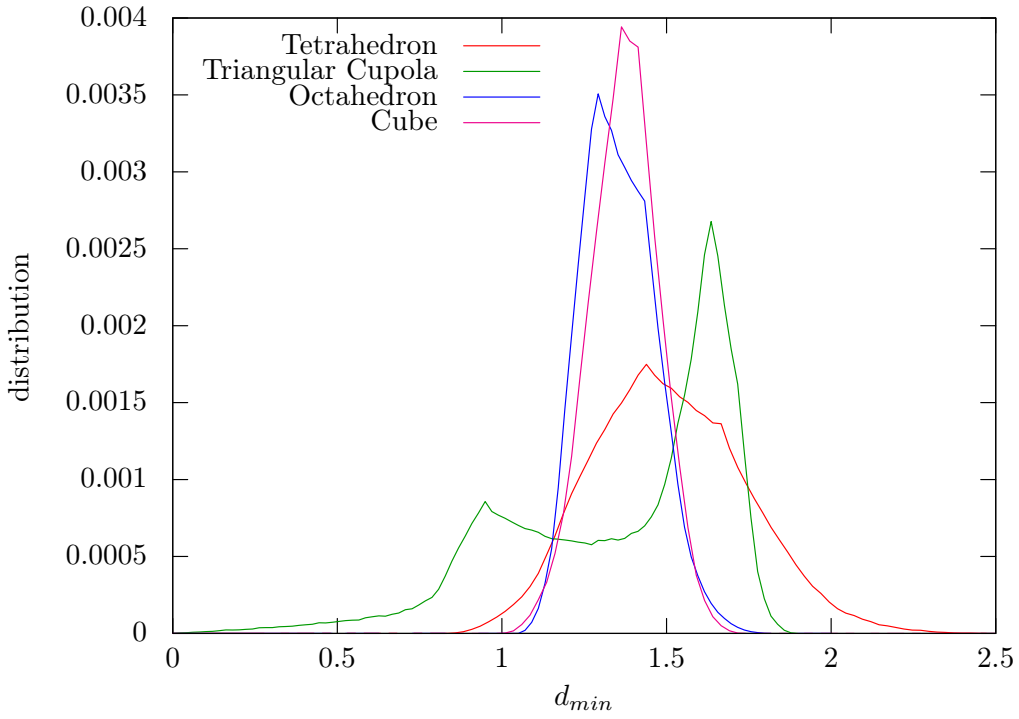


Figure 16: Distribution of *minimum* distances for pairs of Tetrahedra, Triangular Cupola, Octahedra and Cubes with random orientation.

In fig. 18 the results we got from the fits of the SISFs vs the average translation acceptance ratio $\bar{\alpha}$ are plotted, and we rescaled the relaxations times compared to Octahedra to get a global view of the plots for all particles. As can be seen, they agree for all particles and even different packing fractions to a great extent. The plots exhibit a plateau centered at about $\bar{a} \approx 30\%$. Thus, for acceptance ratios in the range of about $20 - 50\%$ similar relaxation times are observed. If the relaxation time is plotted against the maximum displacement, we instead observe a minimum at a maximum displacement corresponding to 30% acceptance. This also means that simulations using such acceptance ratios, will take the least amount of time to finish. It should be noted that we have observed the acceptance ratios for hard particles are quite low for relatively small maximum displacements when compared to other systems, like spheres and Lennard-Jones particle systems. This is to be expected, as was also discussed in the beginning of this section.

We further investigated the dependence of the Mean Square Displacement on the different maximum displacements. An example of the MSD for Octahedra and packing fraction $\phi = 0.62$, can be seen in fig. 19. When scaled by the short time diffusion coefficient, the mean squared displacement for different maximum displacements are in very good agreement, but start to diverge slightly for long times.

Using the findings presented in this section, we decided to use an average acceptance $\bar{a} = 30\%$ for all of our simulations for both translation and rotation moves.

As was mentioned in Section 2, the short time diffusion coefficient can be approximated as in 14, multiplied by the average acceptance $\bar{a}$. Since in our simulations we use N iterations for each MC step and translations have a 50% chance of being attempted, this should be

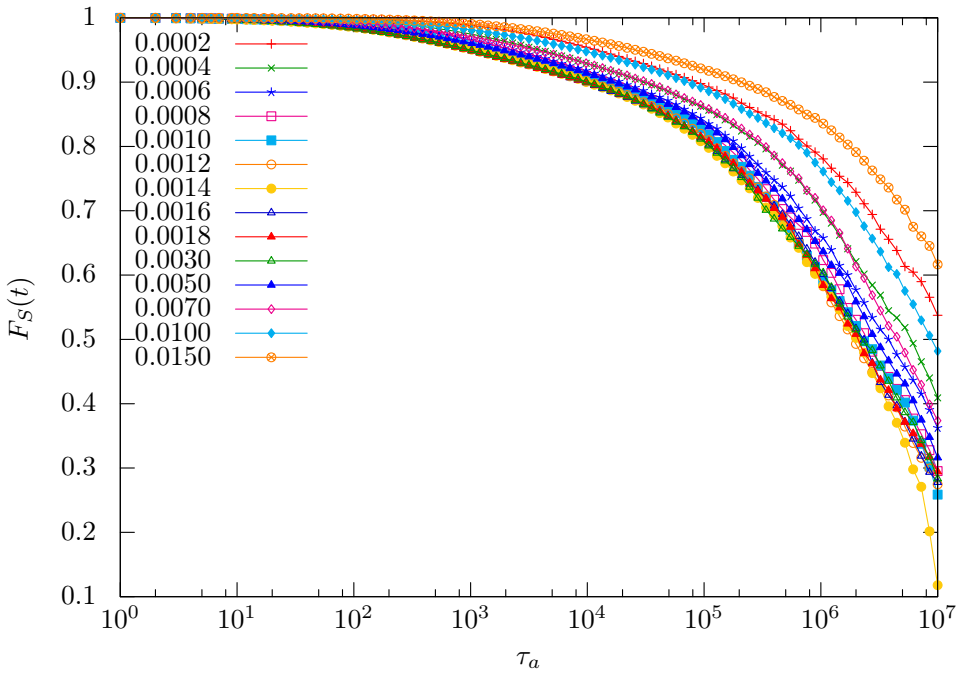


Figure 17: Self-Intermediate scattering function $F_s(t)$ for Octahedra as obtained from MC simulations using varying minimum displacements (as displayed in the figure), for packing fraction $\phi = 0.62$. Maximum displacement units are in fractions of the box length, 14.775.

further multiplied by a factor of 1/2. In principle, if this was not an approximation, the short time diffusion coefficient should agree with the one found by fitting the MSD at short times. In reality though, there is some deviation and we have plotted exactly that in 20. Each point in the plot represents a different maximum displacement, with the smallest maximum displacement close to zero. The larger the maximum displacement, the bigger the disagreement between the two methods of finding the diffusion coefficient. We found though that by giving the acceptance an exponent of ~ 1.4, there is a very good agreement between the two methods even for very large maximum displacements. This exponent is the same, 1.4 for all three particles that were studied, pointing to a similar dependence of the acceptance with respect to the maximum displacement even though they have such a big difference in shape.

It was also observed that picking either a fixed maximum displacement or a fixed acceptance for different packing fractions, the ratio of their relaxation times, agreed to some extend, while for a fixed acceptance, we had the least deviation from the mean ratio among different acceptances. This fact also contributed to us choosing a fixed acceptance across different packing fractions.

3.4 The Self-Intermediate Scattering Function

To extract the SISFs, we performed 5-20 NVT simulations of 2000 particles for various packing fractions. Simulations of up to 10^7 Monte Carlo steps were performed, which took about 2 days on a desktop PC. The SISFs were computed at a wavevector $\mathbf{k}$ close to the first peak of the static structure factor. The results are averaged over all runs that did not crystallize and

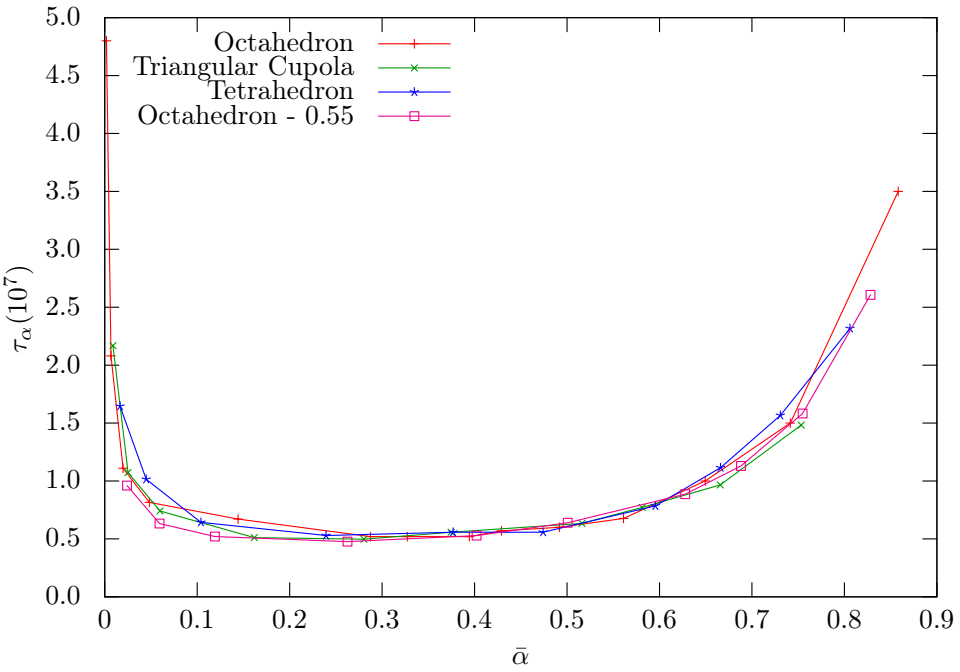


Figure 18: Relaxation time τ_α vs the average acceptance ratio $\bar{\alpha}$ for the three particle shapes studied at packing fraction $\phi = 0.62$ as well as $\phi = 0.55$ for Octahedra. Relaxation times τ_α are rescaled with respect to the relaxation times of Octahedra. For reference, we multiplied the relaxation time for Triangular Cupola by 2.4 and Tetrahedra by 8.4. Octahedra at $\phi = 0.55$ were scaled by 134.0. A plateau in the relaxation time is observed centered at $\bar{\alpha} \approx 30\%$.

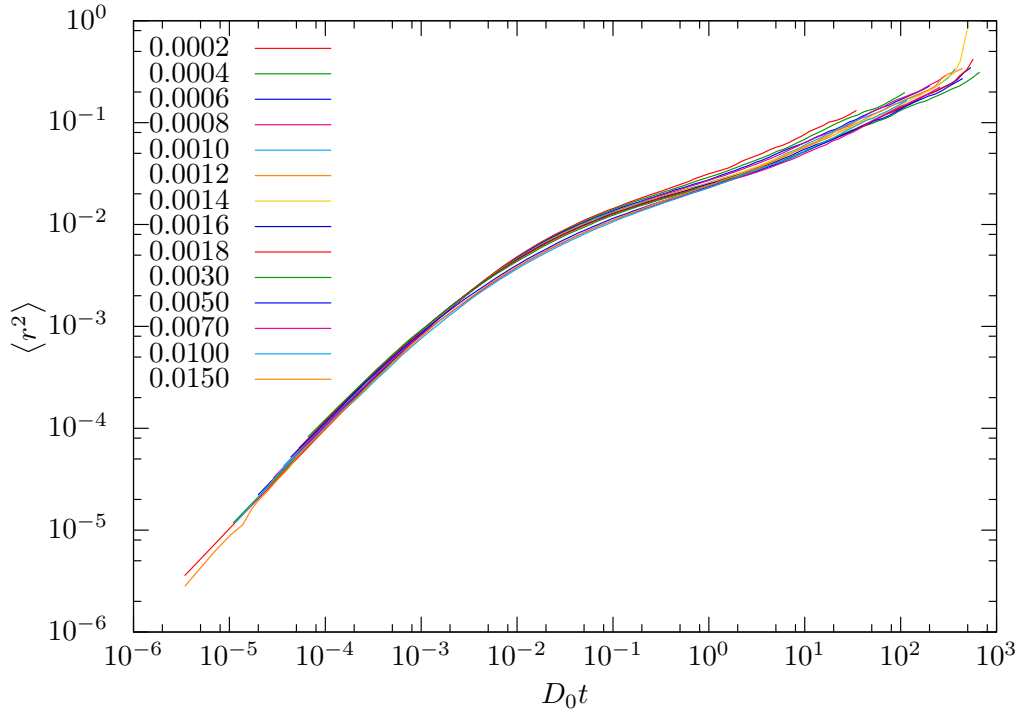


Figure 19: Logarithmic plot of mean squared displacement vs time scaled by the short time diffusion coefficient for Octahedron at 0.62 packing fraction.

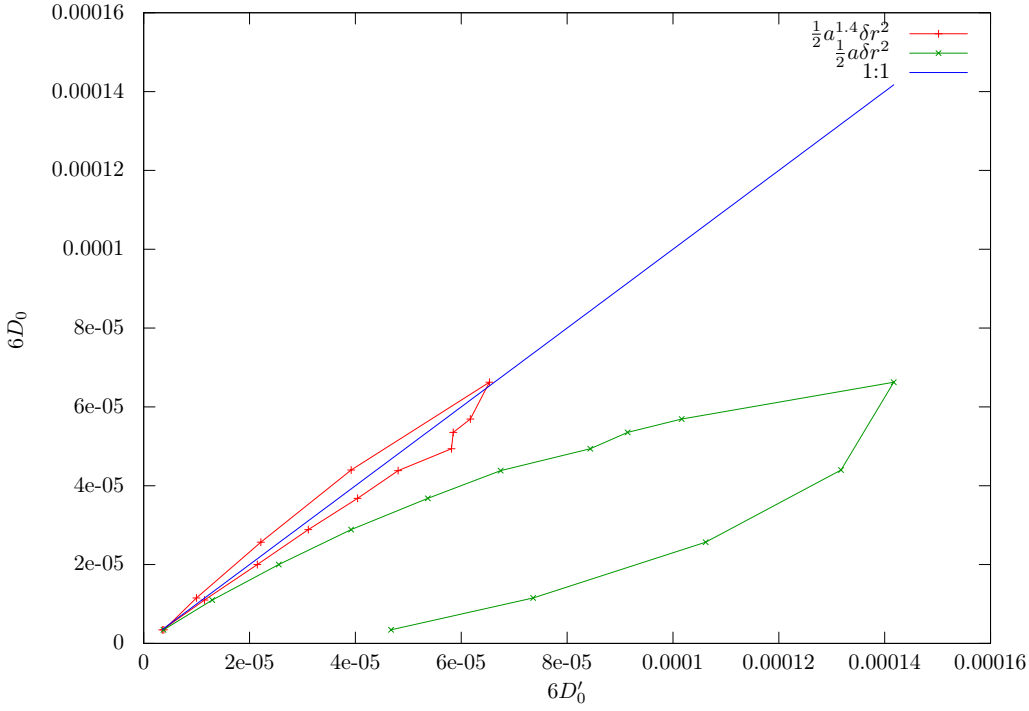


Figure 20: Comparison of the short-time diffusion coefficient acquired by fitting the MSD at short times (y-axis) and by $\frac{1}{2}a\delta r^2$ (x-axis) for Octahedron at 0.62 packing fraction. We have also plotted the last quantity with an exponent of 1.4 of the acceptance. As means of comparison, we have also plotted a 1:1 function.

they are further averaged over different starting times.

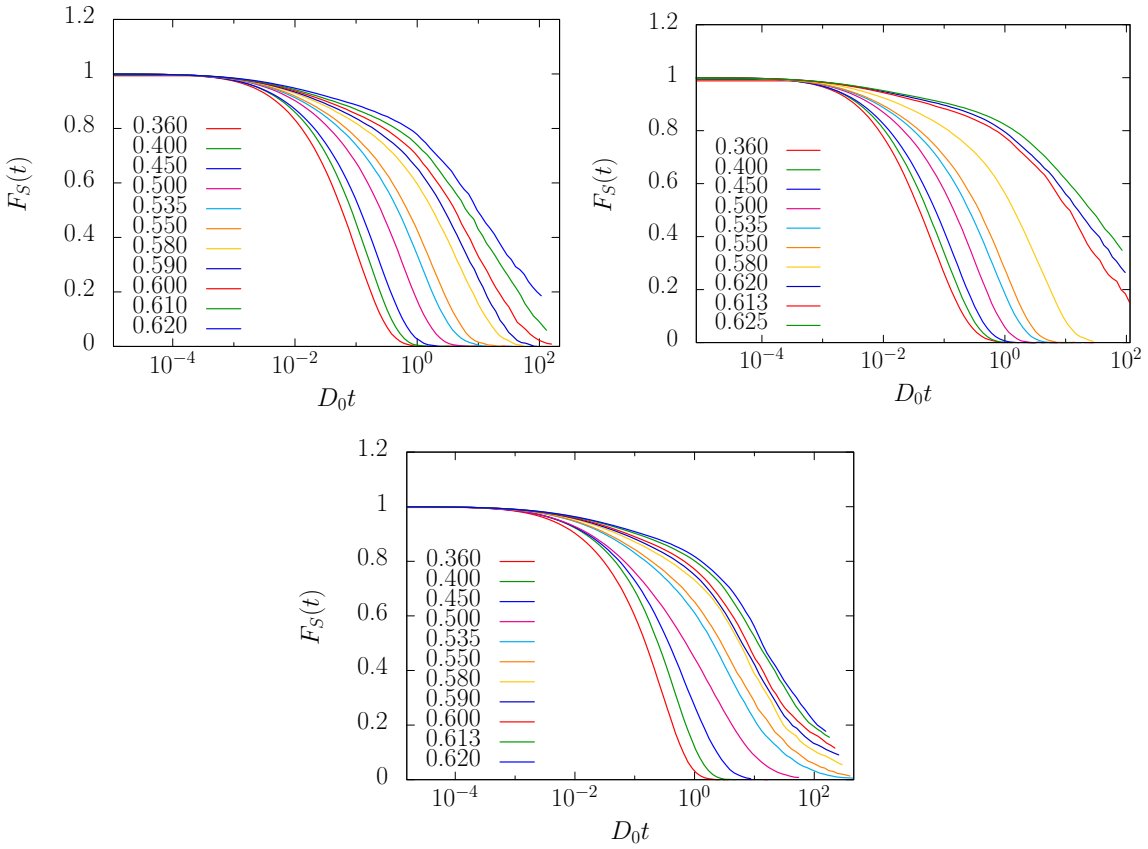


Figure 21: SISFs for packing fraction range $\phi = 0.36$-0.62. The time scale is normalized by the short-time diffusion coefficient, extracted from the mean square displacement. **Top**: Triangular Cupola, Octahedra, **Bottom**: Tetrahedra.

In fig. 21 the resulting SISFs for Triangular Cupola, Octahedra and Tetrahedra, are plotted, where time is scaled by the short-time diffusion coefficient D_0. Triangular Cupola exhibit no peculiarities as is expected since it has no other phases except the isotropic one. We also observe that a plateau is not immediately apparent in the SISF due to the short-time dynamics in MC simulations as already discussed in Section 2. This might also be a consequence of the particles shapes and/or the fact that the SISF expression we used, does not take into account the orientational degrees of freedom of the particles. A slight plateau may be forming though at about $\phi \geq 0.58$. This is further confirmed by the fact that a simple exponential function is not good enough to describe the relaxation. For Octahedra, you may notice a gap between packing fractions $\phi = 0.58$ and 0.62. This is due to the fact that all systems start to crystallize in that packing fraction before reaching the alpha relaxation regime. After 0.62 packing fraction, Octahedra do not crystallize very often, at least in the observable amount of time, but experiments of up to 10^8 steps, seem to suggest that crystallization still occurs, but the nucleation times increase. Again, a plateau may be observed at $\phi \approx 0.58$ and for higher packing fractions we observe a large slowing down of the relaxation compared to Triangular Cupola. Tetrahedra on the other hand exhibit a slightly behavior corresponding to the packing fraction at which a plateau is present in the equation of state. The tails of the SISF are

stretching for $\phi \geq 0.45$, while at $\phi \in [0.55, 0.58]$ a slight plateau is starting to form once again. We were not able to tell if the samples were actually in a crystal state or in the more complex liquid state mentioned earlier since these are hard to tell apart. Fitting the relaxation with a stretched exponential though, exhibited a deviation at the tail, a behavior which is not seen in regular supercooled liquids and is not explained by Mode Coupling Theory. We argue that this effect is due to the formation of clusters of decahedra and icosahedra that was mentioned earlier. The clusters themselves first need to break the cage formed by other clusters and individual particles after which, particles need to break the cage formed by particles from the same cluster. This was confirmed by the means of visualation of the simulation runs. We monitored several clusters which seemed to slowly start moving and then breaking down to their individual constituents. It would be interesting in future work to somehow identify clusters and see how they diffuse before breaking apart.

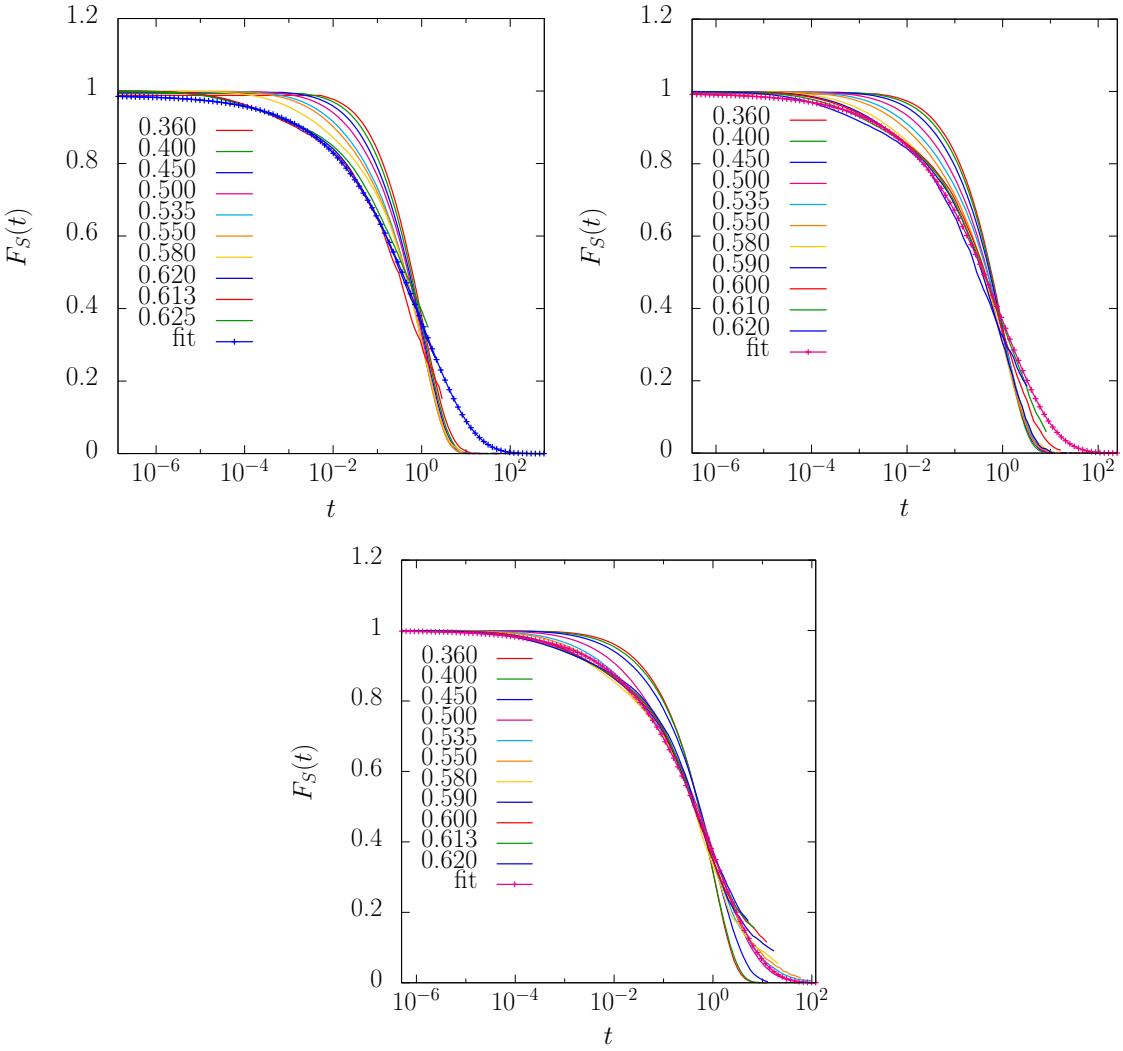


Figure 22: SISFs divided by the relaxation time. The line with points shows the fit of the master curve. **Top**: Triangular Cupola, Octahedra, **Bottom**: Tetrahedra.

In section 2, we mentioned that when the SISF for different packing fractions (or tem-

perature) are divided by their respective relaxation time, MCT predicts they will fall onto a master curve. These figures were plotted in fig. 22 to see if this prediction holds for plyhedra too. For Octahedra and packing fraction $\phi > 0.58$ and Triangular Cupola for $\phi > 0.55$ we see that MCT is more or less confirmed and all curves fall onto a master curve as seen in fig. 22. For Tetrahedra too, as can be seen in fig. 22, curves fall onto a master curve for packing fractions above 0.5 except for the deviation at large times. It can also prematurely be concluded that Octahedra ought to have the highest critical packing fraction ϕ_c with Triangular Cupola second highest and Tetrahedra the lowest. From the above we can say MCT's prediction is mostly confirmed for the three particles investigated. It would be more satisfactory if we had larger packing fractions too, but a lot of time would have been needed to get sufficiently good results.

3.5 Fragility

As was explained in section 2, a glass with a sharp dependence of the relaxation time with temperature, or equivalently inverse packing fraction, is called fragile while in the opposite case, the material is called a strong glass former. In fig. 23 we present the relaxation time vs the packing fraction for the three particles investigated. To describe the data, the VFT equation (eq. (7)) was used and the relaxation time was scaled so that Triangular Cupola and Tetrahedra had the same relaxation time for $\phi = 0$. For reference, the scaling factor κ used for Tetrahedra was $\kappa = 0.71$ and $\kappa = 0.95$ for Triangular Cupola. From fig. 23, it can be seen that the fragility (i.e. the slope of the curves) varies continuously from strong to fragile. For packing fractions $\phi < 0.59$, Octahedra are the strongest and Tetrahedra and most fragile, while for $\phi > 0.59$, and thus closer to the glass transition, Tetrahedra become the strongest glass formers, with Octahedra the most fragile. Triangular Cupola, remain in between the two other particle shapes in fragility.

The data was fit by both a VFT function and the MCT power law, namely $\tau \propto (\phi_c - \phi)^{-\gamma}$ and both gave an equally good approximation to the simulation data. The results of the fits for all three particles can be seen in table 2.

	Fitting Parameters			
Name	ϕ_c^{MCT}	ϕ_c^{VFT}	γ	A
Octahedron	0.655063	0.777219	4.64768	2.5386
Triangular Cupola	0.685377	0.873272	5.42136	4.32045
Tetrahedron	0.796093	1.13021	7.89842	10.8077

Table 2: VFT and MCT power law fitting parameters. ϕ_c^{MCT} is the ergodic-nonergodic transition critical packing fraction, while ϕ_c^{VFT} is a packing fraction extracted from fitting with the VFT and sets the apparent divergence of the relaxation time. The VFT parameter A controls the growth of the relaxation time as the packing fraction approaches ϕ_c^{VFT}.

As expected, the γ exponents are relatively high, indicating that the three glass formers are overall fragile. From the table, we also see that the ergodic-nonergodic transition predicted by MCT lies at higher packing fractions than the ones investigated. For Octahedra, we would need nearly 10^9 MC steps at such packing fractions. This packing fraction is similar for Octahedra and Triangular Cupola, while it is quite a bit higher for Tetrahedra. It should be also noted that Tetrahedra have a larger error due to their peculiar behavior.

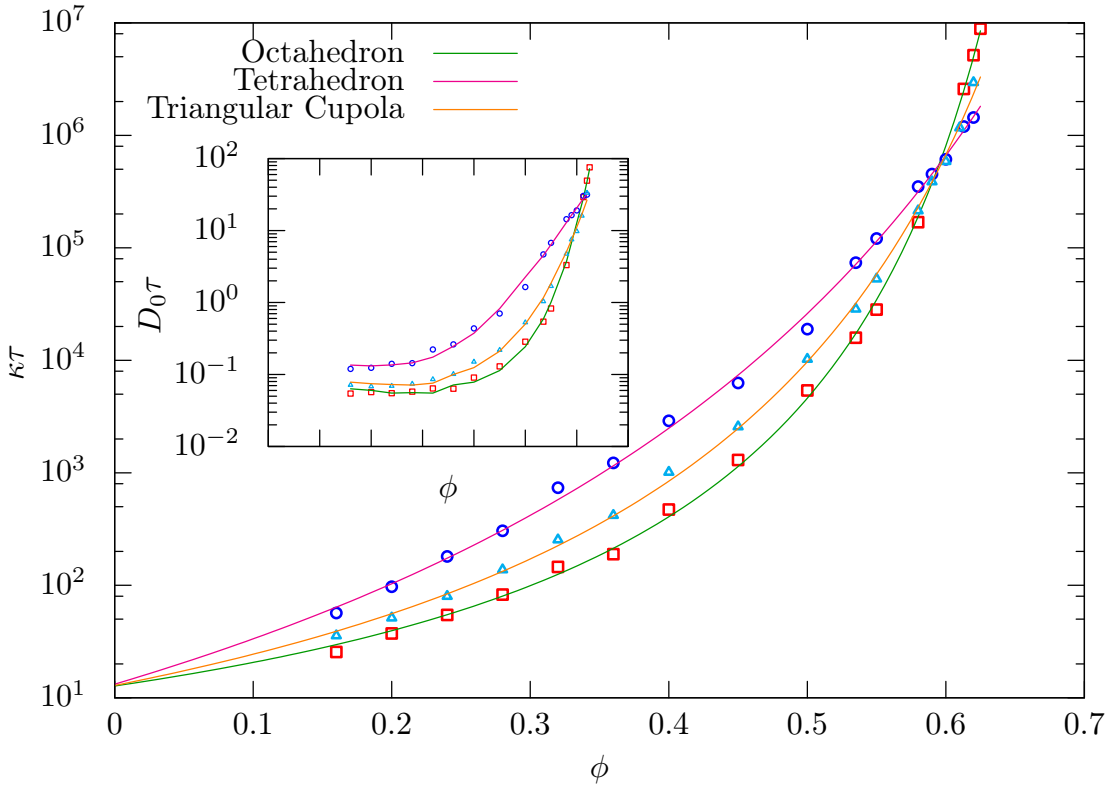


Figure 23: Relaxation time vs packing fraction for the three polyhedra. Relaxation time of Tetrahedra and Triangular Cupola was scaled to all fall on top of each other for small packing fractions. In the inset, the same plot with time scaled by the short-time diffusion coefficient. The lines are VFT fits of the data.

3.6 Octahedron-Tetrahedron Binary Mixtures

Since simulations of glassy dynamics take a lot of time, we decided to explore the phase behavior of binary mixtures of Octahedra and Tetrahedra in our spare time. There has been a lot of work done in the behavior of single compounds but mixtures of polyhedra is still a largely unexplored field. The parameter space from going from a single compound to a binary mixture, grows tremendously. One can change the relative size between the two particles and the ratio of the number of particles.

We chose to investigate for fixed particle sizes how the ratio of their numbers changes the behavior of the mixture. A binary mixture of Tetrahedra and Octahedra was chosen because Tetrahedra and Octahedra in a 2:1 ratio are space filling [54]. An example of this space filling structure can be seen in fig. 24 which was generated using the floppy box Monte Carlo method. The lattice vectors for the unit cell of the space filling structure of unit volume are: $\mathbf{v}_1 = (0.7507, 0.0, 0.0)$, $\mathbf{v}_2 = (0.2451, 0.7097, 0.0)$, $\mathbf{v}_3 = (0.6648, 0.1788, 0.8332)$. Both Octahedra and Tetrahedra have distinct crystal phases which were described in earlier

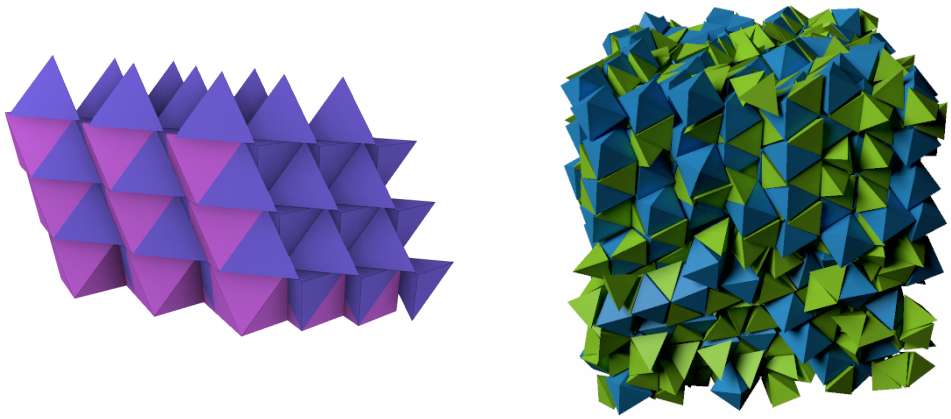


Figure 24: **Left**: Space filling structure obtained from floppy box MC. The unit cell contains 2 Tetrahedra and 1 Octahedron. In this figure, the unit cell is repeated 3 times in each direction of the lattice vectors. **Right**: Snapshot of configuration at pressure $\beta P = 46.5$ and packing fraction $\phi = 0.7$.

sections. A mixture of Tetrahedra and Octahedra is also interesting because a combination of them can form larger Octahedra and Tetrahedra. We therefore investigate how the ratio of the two particles can be used to fine-tune the different crystal phases in the mixture. We know of only one published work on binary mixtures of polyhedra, which was published very recently by Escobedo et al. [55]. They studied binary mixtures and more specifically space tessellating compounds as well as the binary mixture of Octahedra and Tetrahedra. The work presented here, is by no means extensive and/or final but more of a proof of concept for future work.

To extract the equations of state for the various ratios, NPT MC simulations were performed by compressing a low density system with a packing fraction ϕ of about 0.1. We only used $6 \cdot 10^5$ MC steps per pressure point with $2 \cdot 10^5$ MC steps for equilibration. At least 10^6 steps are usually recommended for such runs though. Additionally, 2000 particles were used in the simulations and the equations of state were extracted for particle ratios of 2:18

to 18:2. The results are depicted in fig. 25. The pressure in the figure was normalized by

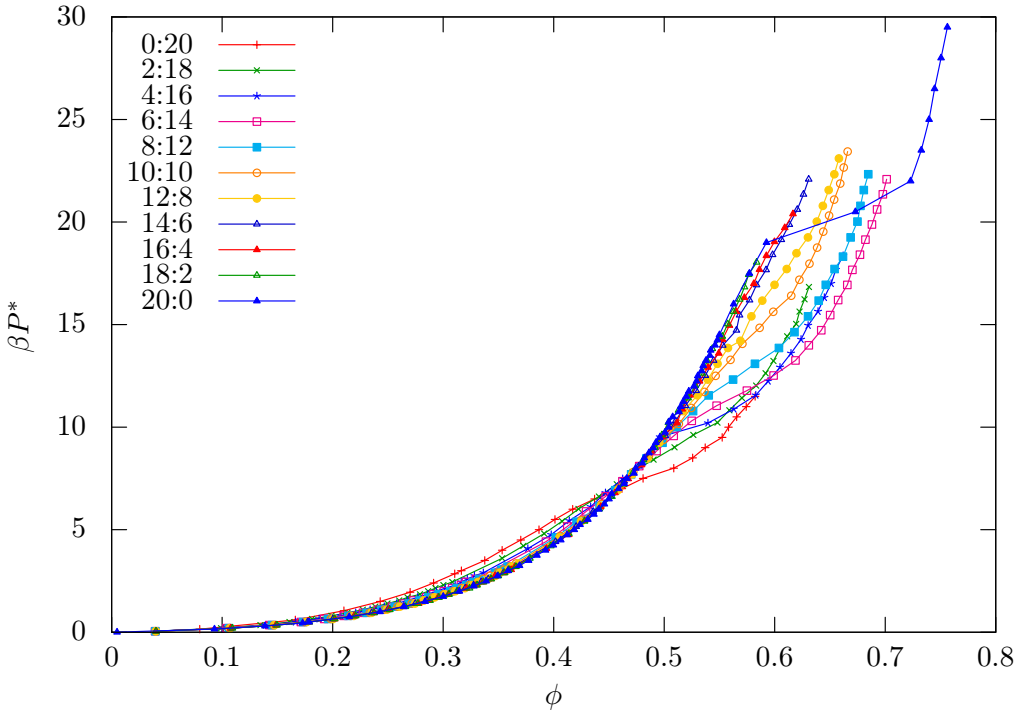


Figure 25: Equations of state for various ratios of Octahedra and Tetrahedra. In the figures, the ratios are of the type N_o:N_t, with 'o' for Octahedron and 't' for Tetrahedron. In the right figure, pressure is multiplied by the average volume of the particles, i.e. by $\frac{N_oV_o+N_tV_t}{N_o+N_t}$.

the average volume of a particle $\frac{N_oV_o+N_tV_t}{N_o+N_t}$, which gives a better picture of the step seen in the packing fraction for scaled pressures of about 28. We see that the step in the equation of state which was already present in Tetrahedra, becomes more pronounced for the ratios 4:16, 6:14 and 8:12 and after 10:10 the step starts fading out. We should mention that this step usually seen for pure Tetrahedra, is a different transition from the one mixtures with ratios close to 1:2 experience, which is the space filling structure. The pure Octahedra, were also the only "mixture" which crystallized to the Minkowski crystal. We see that the equations of state become more sensitive to pressure change as the number of Tetrahedra changes. From these results, it is apparent that the mixture's compressibility and the existence of a phase transition can be tuned by changing the ratio of the particle numbers.

For the mixture with ratio 6:14 we have included a snapshot of the configuration at pressure 46.5 and a packing fraction of 0.7 in fig. 24. This ratio is the closest to the space filling ratio of 1:2. Many clusters of the space filling structure and other similar structures are seen to form, while it seems that a perfect crystal is difficult to form. These clusters are also present in other ratios in varying numbers.

We do think that this small project could be further extended. A more precise investigation with more steps and higher pressures needs to be conducted. The investigation can also be done for different edge sizes for each particle and thus a phase diagram could be build. As was mentioned, the parameter space is very large and thus one would have to limit himself for example to space filling mixtures, or particles that can be synthesized in the lab for

some fruitful results. All in all, we think that there will be an increased interest in mixtures consisting of plyhedra in the near future because of their flexibility, tunability and simplicity.

4

Conclusions and Outlook

In the previous section we demonstrated our results from simulations of hard polyhedra. We first investigated the general properties and structural information of the three particles. Here it was found that Triangular Cupola tend to align their hexagonal faces with increasing packing fraction, while Tetrahedra showed some peculiar behavior owning to the formation of clusters of decahedra and icosahedra. For some time it was accepted that Monte Carlo simulations, are not fit for simulations of dynamics, but it was shown recently [51] and predicted by MCT [44] that they can be used to some extent. A series of experiments where conducted, and it was found that the relaxation time behavior with respect to the average acceptance of translations, similar for all three particles, shows a plateau in the range of 20-50%. This analysis was similar to the one done in [51], and we chose to use a 30% acceptance for our simulations. Having established the general methodology for our simulations, the SISFs were calculated for various packing fractions. The expected plateau that is generally seen in glass formers was not apparent most probably due to the nature of the short-time dynamics of MC and the shapes of the particles. Again, Tetrahedra showed some unexpected behavior, deviating from MCT. This was also attributed to the formation of clusters of decahedra and icosahedra. We had a difficult time to get the SISF for packing fractions of around 0.58-0.62 for Octahedra because of the crystallization that occurred. MCT predicts the SISF falling onto a master curve when divided by the relaxation time up to first approximation in $1/(\phi - \phi_c)$ and this was confirmed with respect to packing fraction, with the exception of Tetrahedra deviating at large times. In the end, we made an Arrhenius plot of the relaxations times extracted from fitting the SISFs with a stretched exponential, which allowed us to say something about the fragility of the three polyhedra. Octahedra were found to be the most fragile glass formers while Tetrahedra seemed to be the least fragile ones for packing fractions close to the glass transition. These curves were also fitted using a power law predicted by MCT and found the ergodic-nonergodic critical packing fraction for each. These packing fractions, are generally high compared to other well known systems for which packing fractions $\phi_c \approx 0.57$-0.59 are reported [56].

We additionally looked briefly at a binary mixture of Octahedra and Tetrahedra of equal edge length. We showed that the compressibility and existence of a phase transition can be tuned by changing the concentration of each particle.

In conclusion, we think we were able to build high performing and reliable Monte Carlo simulations of convex polyhedra and managed to confirm several predictions of MCT. We discovered various properties of the glassy behavior of these particles, something that was not previously attempted. We like to think this Thesis can be extended in many ways given the appropriate amount of time. To make the results presented here, even more concise, we suggest a cross-validation with results from MD simulations as well as a comparison with real experiments. The methodology presented here can also be further extended to poly-disperse systems and to additional polyhedron types and setups. With our current configuration, simulations of 10^8 MC steps are also possible and could provide additional insight, especially at higher packing fractions. Unfortunately, reaching even longer times seems very difficult with

current algorithms. Additionally, it could be interesting to investigate the kinetic heterogeneities exhibited by these systems. Glassy behavior is not yet well understood and any well structured analysis of this behavior takes a lot of time. Hopefully, the suggestions provided could help get a better understanding especially in the systems we studied, since there is an absence of interactions.

In the last part of our results, we also briefly showed the potential of binary systems. Binary systems of polyhedra are a step up in complexity but the complexity in behavior may be even larger. The possibilities here are many, one could create phase diagrams for different concentrations of the particles. A different approach would be to do the same investigation using different ratios of edge lengths.

With the advancement seen in nano-synthesis, polyhedra have drawn increasing interest. We think there are many advancements to be made in the near future and we hope we have made a contribution to the further understanding of these simple yet complex systems.

A

Visualization Methods

For the purpose of visualizing the various configurations we obtained from simulations, we developed several tools. Visualization provides us with a more concrete image of what is really happening beside the aesthetics, and should be part of the *toolset* of every scientist. Below, these tools are presented, and can be accessed on GitHub at https://github.com/Grieverheart. At the moment, the tools are not very user friendly and have a few dependencies which can be difficult to assemble for inexperienced users, but if there is enough interest, they can be made more accessible.

OpenGL (Open Graphics Library) is a cross-language, multi-platform API for rendering 2D and 3D computer graphics. The API is typically used to interact with a GPU, to achieve hardware-accelerated rendering.

Inspired by the work done by our former group member Michiel Hermes, we developed a tool for visualizing particle configurations from text files in 3D. OpenGL 2 was used for compatibility with older systems. In addition we used the freeglut library for cross-platform compatibility. In addition to Michiel's implementation, some additional functionality was added. This includes the following:

- Animation by providing a series of configurations files
- Particle selection and hiding
- Orthographic and Orthogonal Projections
- Resizing particles
- Calculation of Bond Order Parameter and visualization of solid-like particles
- Real-Time diffraction patterns using a fast Fourier transform (FFTW library)
- Constructive Solid Geometry (CSG) for visualization of free space (OpenCSG library)

An example of the visualization of free volume with the use of CSG can be seen in fig. 26. The size of the box used to slice through the material, can be adjusted.

For more visually pleasing visualizations, we additionally developed a similar tool using OpenGL 3.3 but omitted the functionality provided by the previous program. An example of the visual representation provided by this implementation, can be seen in fig. 14. To achieve this result, we have used what is called screen space ambient occlusion. The visual quality as well as the particle and background colors can be tuned according to one's liking through a graphical user interface.

These tools can also be used in conjunction with simulations to make them interactive for either educational or practical purposes. On GitHub, one can also find a library we developed in C++ for creating floppy-box MC simulations. The library is easy to use and quite flexible. We have used this library together with the visualization tool to create such interactive simulation, an example video of which can be found at http://youtu.be/aat13SDFDQI .

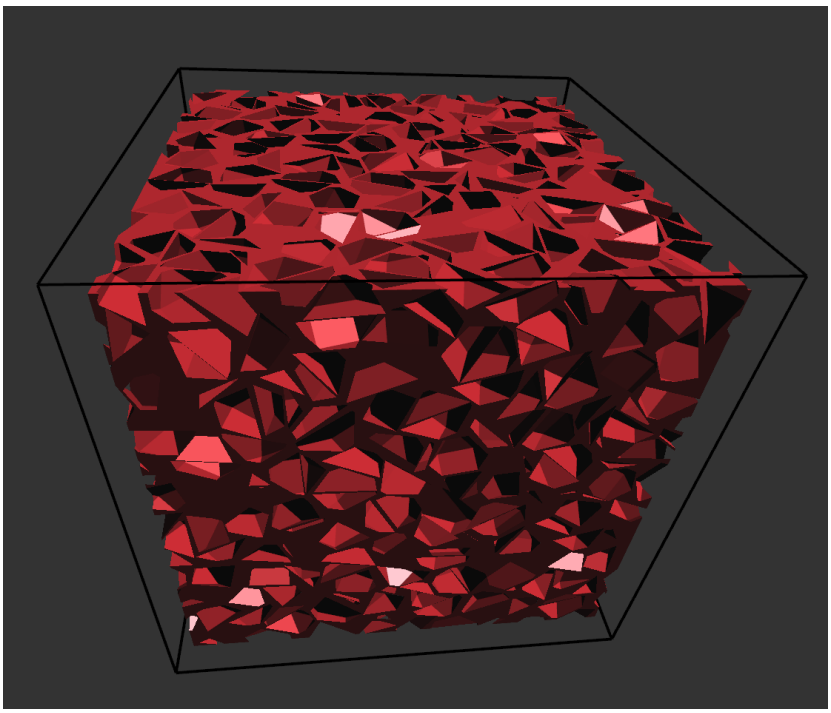


Figure 26: An example of the visualization of the free volume in a particle configuration of Triangular Cupola.

References

[1] Zallen, R. *The Physics of Amorphous Solids*; Physics textbook; John Wiley & Sons, 1998.

[2] Adam, G.; Gibbs, J. H. *The Journal of Chemical Physics* **1965**, *43*, 139–146.

[3] Gibbs, J. H.; DiMarzio, E. A. *The Journal of Chemical Physics* **1958**, *28*, 373–383.

[4] Ngai, K. L. *Journal de Physique* **1992**, *2*, 61.

[5] Götze, W. *Liquids, Freezing and the Glass Transition : Part I (Les Houches Summer School Proceedings)*; North-Holland, 1991.

[6] Götze, W.; Sjögren, L. *Reports on Progress in Physics* **1992**, *55*, 241.

[7] Weisstein, E. W.; *"Convex Polyhedron." From MathWorld–A Wolfram Web Resource.* `http://mathworld.wolfram.com/ConvexPolyhedron.html`.

[8] Tao, A.; Sinsermsuksakul, P.; Yang, P. *Angewandte Chemie International Edition* **2006**, *45*, 4597–4601.

[9] Henzie, J.; Grünwald, M.; Widmer-Cooper, A.; Geissler, P. L.; Yang, P. *Nature Materials* **2012**, *11*, 131–137.

[10] Zhang, J.; Luo, Z.; Martens, B.; Quan, Z.; Kumbhar, A.; Porter, N.; Wang, Y.; Smilgies, D.-M.; Fang, J. *Journal of the American Chemical Society* **2012**, *134*, 14043–14049.

[11] Eguchi, M.; Mitsui, D.; Wu, H.-L.; Sato, R.; Teranishi, T. *Langmuir* **2012**, *28*, 9021–9026.

[12] Quan, Z.; Siu Loc, W.; Lin, C.; Luo, Z.; Yang, K.; Wang, Y.; Wang, H.; Wang, Z.; Fang, J. *Nano Letters* **2012**, *12*, 4409–4413.

[13] Choi, J. J.; Bian, K.; Baumgardner, W. J.; Smilgies, D.-M.; Hanrath, T. *Nano Letters* **2012**, *12*, 4791–4798.

[14] Zhang, Y.; Lu, F.; van der Lelie, D.; Gang, O. *Phys. Rev. Lett.* **2011**, *107*, 135701.

[15] Rossi, L.; Sacanna, S.; Irvine, W. T. M.; Chaikin, P. M.; Pine, D. J.; Philipse, A. P. *Soft Matter* **2011**, *7*, 4139–4142.

[16] de Graaf, J.; Filion, L.; Marechal, M.; van Roij, R.; Dijkstra, M. *The Journal of Chemical Physics* **2012**, *137*, 214101.

[17] Tao, A. R.; Ceperley, D. P.; Sinsermsuksakul, P.; Neureuther, A. R.; Yang, P. *Nano Letters* **2008**, *8*, 4033–4038; PMID: 18928325.

[18] Sun, Y.; Xia, Y. *Science* **2002**, *298*, 2176–2179.

[19] Matijevic, E. *Accounts of Chemical Research* **1981**, *14*, 22–29.

[20] Xia, X.; Xia, Y. *Nano Letters* **2012**, *12*, 6038–6042.

[21] Wiley, B.; Herricks, T.; Sun, Y.; Xia, Y. *Nano Letters* **2004**, *4*, 1733–1739.

[22] Barnard, A. S.; Lin, X. M.; Curtiss, L. A. *The Journal of Physical Chemistry B* **2005**, *109*, 24465–24472; PMID: 16375449.

[23] Zhang, X.; Dong, C.; Zapien, J.; Ismathullakhan, S.; Kang, Z.; Jie, J.; Zhang, X.; Chang, J.; Lee, C.-S.; Lee, S.-T. *Angewandte Chemie International Edition* **2009**, *48*, 9121–9123.

[24] Wu, H.-L.; Kuo, C.-H.; Huang, M. H. *Langmuir* **2010**, *26*, 12307–12313.

[25] Solomon, M. J.; Glotzer, S. C. *Nature Materials* **2007**, *6*, 557–562.

[26] Damasceno, P. F.; Engel, M.; Glotzer, S. C. *Science* **2012**, *337*, 453–457.

[27] de Graaf, J.; van Roij, R.; Dijkstra, M. *Phys. Rev. Lett.* **2011**, *107*, 155501.

[28] Torquato, S.; Jiao, Y. *Nature* **2009**, 876–879.

[29] Haji-Akbari, A.; Engel, M.; Keys, A. S.; Zheng, X.; Petschek, R. G.; Palffy-Muhoray, P.; Glotzer, S. C. *Nature* **2009**, 773–777.

[30] Khadilkar, M. R.; Escobedo, F. A. *The Journal of Chemical Physics* **2012**, *137*, 194907.

[31] Manousiouthakis, V. I.; Deem, M. W. *The Journal of Chemical Physics* **1999**, *110*, 2753–2756.

[32] Newman, E.; Barkema, G. *Monte Carlo Methods in Statistical Physics*; Clarendon Press, 1999.

[33] Filion, L.; Marechal, M.; van Oorschot, B.; Pelt, D.; Smallenburg, F.; Dijkstra, M. *Phys. Rev. Lett.* **2009**, *103*, 188302.

[34] Frenkel, D.; Smit, B. *Understanding Molecular Simulation: From Algorithms to Applications*; Computational Science Series, Vol 1; Acad. Press, 2002.

[35] Marsaglia, G. *Ann. Math. Stat.* **1972**, *43*, 645–646.

[36] van den Bergen, G.; *A Fast and Robust GJK Implementation for Collision Detection of Convex Objects*; 1999. `www.win.tue.nl/ gino/solid/jgt98convex.pdf`.

[37] Bittle, W.; *GJK (Gilbert-Johnson-Keerthi).* `http://www.codezealot.org/archives/88`.

[38] Van Den Bergen, G. *Collision Detection in Interactive 3D Environments*; The Morgan Kaufmann Series in Interactive 3D Technology; Morgan Kaufman Publishers, 2004.

[39] Bengtzelius, U.; Götze, W.; Sjolander, A. *Journal of Physics C: Solid State Physics* **1984**, *17*, 5915.

[40] Reichman, D. R.; Charbonneau, P. *Journal of Statistical Mechanics: Theory and Experiment* **2005**, *2005*, P05013.

[41] Angell, C. A.; Ngai, K. L.; McKenna, G. B.; McMillan, P. F.; Martin, S. W. *Journal of Applied Physics* **2000**, *88*, 3113–3157.

[42] Steinhardt, P. J.; Nelson, D. R.; Ronchetti, M. *Phys. Rev. B* **1983**, *28*, 784–805.

[43] Kawasaki, K. *Phys. Rev.* **1966**, *150*, 291–306.

[44] Kob, W. *The Mode-Coupling Theory of the Glass Transition*; ACS Publicationsy, 1997; Chapter 4, pp 28–44.

[45] van Megen, W.; Underwood, S. M. *Phys. Rev. E* **1994**, *49*, 4206–4220.

[46] Kob, W.; Andersen, H. C. *Phys. Rev. E* **1995**, *52*, 4134–4153.

[47] Rapaport, D. *The Art of Molecular Dynamics Simulation*; Cambridge University Press, 2004.

[48] Whitelam, S.; Geissler, P. L. *The Journal of Chemical Physics* **2007**, *127*, 154101.

[49] Branka, D. M. H. A. C. *Molecular Physics* **1998**, *94*, 447–454.

[50] Sanz, E.; Marenduzzo, D. *The Journal of Chemical Physics* **2010**, *132*, 194102.

[51] Berthier, L.; Kob, W. *Journal of Physics: Condensed Matter* **2007**, *19*, 205130.

[52] Einstein, A. *Annalen der Physik* **1905**, *322*, 549–560.

[53] Ni, R.; Gantapara, A. P.; de Graaf, J.; van Roij, R.; Dijkstra, M. *Soft Matter* **2012**, *8*, 8826–8834.

[54] Deza, M.; Shtogrin, M. *European Journal of Combinatorics* **2000**, *21*, 807 – 814.

[55] Khadilkar, M. R.; Escobedo, F. A. *The Journal of Chemical Physics* **2012**, *137*, 194907.

[56] Brambilla, G.; El Masri, D.; Pierno, M.; Berthier, L.; Cipelletti, L.; Petekidis, G.; Schofield, A. B. *Phys. Rev. Lett.* **2009**, *102*, 085703.