Universiteit Utrecht

Faculty of Science
Debye Institute

Roadmap to the Nucleation of ico-AB$_{13}$ Binary Crystals Using the Seeding Approach and Machine Learning

Master Thesis

Author:
Jayden Peter Savage
6150985
Experimental Physics

Supervisor:
Prof. Dr. M. Dijkstra.
Utrecht University

Daily Supervisor
Mr. G. M. Coli.
Utrecht University

30 June 2019

Abstract

The equations of state for AB_{13} crystalline and binary fluid WCA hard spheres were determined for varying temperatures in order to define the pressure $\beta P\sigma_L^3$ and the number density $\rho\sigma_L^3$ regions at which coexistence occurs. The pressure of coexistence of the two phases was then calculated by means of thermodynamic integration for a temperature $k_BT\sigma_L^3 = 0.025$ and found to be $\beta P\sigma_L^3 = 45.38$. It was then necessary to differentiate the local structure of particles in order to define and track the nucleation of AB_{13} crystals within a binary WCA fluid, thus the bond order parameters were investigated in order to make this classification. It was found that we were able to classify the particles as either crystalline species A or B, crystalline FCC or fluid phase using eight-fold and six-fold symmetries. A neural network ensemble was also developed to classify the local structure of particles as well as determine the feature importance. The feature importance corroborated our findings of the importance of the eight-fold and six-fold symmetries. Finally nucleation of the AB_{13} crystalline phase was achieved for a temperature $k_BT/\epsilon = 0.025$ and pressure $\beta P\sigma_L^3 = 62.00$ using the seeding approach to nucleation.

Contents

1 Introduction

One of the most interesting phenomena to be observed in condensed matter systems is that of nucleation, the abrupt structural change observed to occur at specific pressures and temperatures [1]. Understanding nucleation has proven to be instrumental to the development of industry, medicine (how drugs nucleate in the blood) and even meteorology (nucleation in atmospherics). In this study we focus on the nucleation of AB_{13} crystals surrounded by a binary fluid using computer simulations[i]. We make use of computer simulations as we aim to obtain the physical processes under investigation in [2, 3, 4]. Binary compounds with a stoichiometric ratio AB_{13} are of particular interest due to their potential applications to industry. As seen in [5], the negative thermal expansion (NTE) properties of La(Fe, Se)$_{13}$ based compounds suggest their potential application as NTE materials. Additionally, in [6], it was found that $LaCo_{13}$ posses strong ferromagnetic properties. Ferromagnets have a plethora of industrial applications from transformers and magnetic recording tape to electromagnets.

There are two main types of AB_{13} crystal structure; namely, icosahedral and cuboctahedral AB_{13} crystal structure abbreviated as *ico*-AB_{13} and *cub*-AB_{13} respectively. As seen in [7], *cub*-AB_{13} is metastable with respect to *ico*-AB_{13} for a binary mixture of colloidal hard spheres, therefore we draw our attention to the nucleation of *ico*-AB_{13} crystal structures. The unit cell of *ico*-AB_{13} consists of eight simple cubic subcells, species A particles, containing icosahedral clusters of 13 species B particles which are rotated by $90^{\cdot}$ between each cubic subcell [8].

The specifics of the system that we simulate is outlined in further detail in section 3. Before nucleation can be achieved the regime at which the binary fluid phase coexists with the AB_{13} crystal phase needs to be determined, this is achieved by calculating the equations of state (EOS) in computer simulations. The EOS are given in section 4 for multiple temperatures $k_BT/\epsilon = 0.025$, $k_BT/\epsilon = 0.1$ and $k_BT/\epsilon = 0.2$; consequently, the corresponding pressure P and density ρ regimes at coexistence are given. The free energy of AB_{13} has been calculated between the crystal and fluid phases by means of thermodynamic integration, and from these calculations the exact pressure at which coexistence occurs can be determined for a given temperature k_BT/ϵ and size ratio α between A and B species, the results obtained from the thermodynamic integration can be found in section 4.

To monitor the nucleation of AB_{13} crystals a characterization of local structure is required. We investigate whether local bond order parameters - introduced in section 2.2 are sufficient to classify the local structure of a colloidal particle. In section 5.1 we present the results of the most effective bond order parameters investigated. We then explore an alternative method to classify the local structure of a particle, whereby we make use of an artificial neural network. We investigate the best possible neural network architecture and optimizer for the task of local structure identification using the bond order parameters as features of the network. Artificial neural networks are introduced in section 2.3, the success

[i] Simulations were conducted in the C programming language and later translated into C++ as C++ offers more complex structures and hence more suitable for the task.

of the neural networks are then presented in section 5.2. Using the best neural network performer we then determine feature importance using the feature extraction algorithm proposed in [9]. Once we are able to classify the local structure of a colloidal particle we are then able to identify particles that belong to an AB_{13} crystal in a binary fluid based on a clustering algorithm (see section 2.4). We then attempt to achieve nucleation of AB_{13} crystals using the seeding approach as outlined in [10], this method is summarized in section 2.4. A trajectory of seed growth is presented in section 6
Finally, we present our conclusions in section 7; additionally, we provide suggestions for further research that build on the investigations contained in this paper.

2 Theory

In this section we provide a brief introduction to concepts necessary to understand the investigations undertaken in this paper. As mentioned within the introduction, algorithms used in the computer simulations are reserved for the appendix as they are not necessary to understand the system under investigation.

2.1 Classical Nucleation Theory

There are two main types of nucleation, namely heterogeneous nucleation and homogeneous nucleation [1]. The main distinction between these two types of nucleation is that heterogeneous nucleation is the crystallization of a substance with the aid of an impurity, while homogeneous nucleation occurs without any impurities. For the purposes of this study we will focus on homogeneous nucleation. To understand Classical Nucleation Theory in the context of homogeneous nucleation we consider a system in which an on average spherical nucleus of the new phase, called the "daughter phase"; is contained in a bulk, called the "parent phase".

In order for the metastable parent phase to transition to the stable daughter phase a nucleation barrier must be overcome. This barrier can best be described by the free energy of the nucleus, which can by given by the sum of the free energy associated with a bulk term that scales with the volume of the nucleus, and the free energy associated with the interface between the parent phase and the daughter phase.

$$\Delta G = -\frac{4}{3}\pi r^3 \Delta\mu + 4\pi r^2 \gamma. \tag{1}$$

where r represents the radius of the nucleus, γ the surface tension between the nucleus and the bulk, and $\Delta\mu$ is given by $\Delta\mu = \mu^{parent} - \mu^{daughter}$, the chemical potential difference between the parent and the daughter phase.

The free energy 1 is graphically shown in Figure 1 as a function of the radius r. One can see that the free energy barrier labeled ΔG^*, is a maximum at the critical radius labeled r_c. Should the radius of the nucleus be larger than the critical radius it is expected that the nucleus will grow; conversely, should the radius of the nucleus be less than the critical radius then the nucleus will shrink [1].

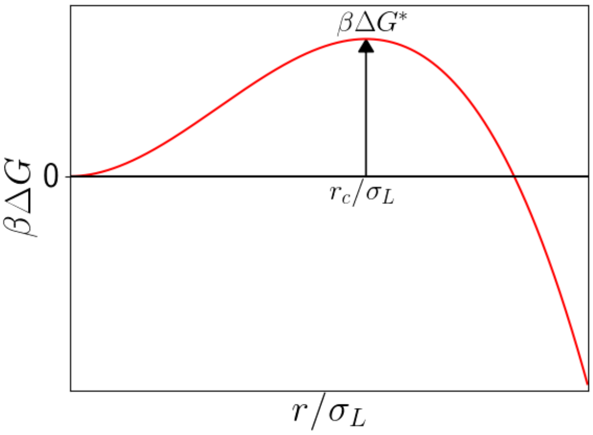


Figure 1: The Gibbs free energy for the formation of a nucleus of radius r/σ_L with a nucleation barrier height denoted by $\beta\Delta G^*$ and the critical radius labeled r_c.

In order to determine the critical radius r_c one solves the case that $\frac{d\Delta G}{dr} = 0$. The critical radius size is thus given by

$$r_c = \frac{2\gamma}{\Delta\mu}. \tag{2}$$

Using this critical radius in equation 1, we find that the free energy barrier is given by

$$\Delta G^* = \frac{16}{3}\frac{\pi\gamma^3}{\Delta\mu^2} \tag{3}$$

2.2 Local bond-order parameters

Before we introduce local bond-order parameters we will take a short excursion to introduce radial distribution functions, which can be used to define a cut-off radius necessary to determine the nearest neighbors of a particle i which will be used to calculate the local bond-order parameters. The radial distribution function denoted $g(r)$ gives the probability distribution to find a particle a distance r from a particle at the origin in a homogeneous isotropic fluid phase and is defined as

$$g(r) = \frac{\rho^{(2)}(\mathbf{r},\mathbf{r}')}{\rho^2} \tag{4}$$

where $\rho = N/V$, $r = |\mathbf{r} - \mathbf{r}'|$ and $\rho^{(2)}(\mathbf{r},\mathbf{r}')$ denotes the two-body correlation function given by

$$\rho^{(2)}(\mathbf{r},\mathbf{r}') = \left\langle \sum_{i=1}^{N}\sum_{j=1}^{N} \delta(\mathbf{r} - \mathbf{r}_i)\delta(\mathbf{r}' - \mathbf{r}_j) \right\rangle. \tag{5}$$

The brackets $\langle \cdots \rangle$ denote an ensemble average. From the radial distribution function $g(r)$ the first observed peak will denote the first shell of particles around particle i, and in turn will define our cut-off radius necessary to determine the nearest neighbors of particle i.

As seen in [11, 12] and as introduced by ten Wolde [13], local bond-order parameters can be used in order to quantify the local structures in disordered systems; consequently, allowing for the identification of solid like and liquid like particles.
The bond order parameter for particle i is defined as the sum of spherical harmonics $Y_{lm}(\mathbf{r}_{ij})$, over all neighbours j; and is given by

$$q_{lm}(i) = \frac{1}{N_b(i)} \sum_{j=1}^{N_b(i)} Y_{lm}(\mathbf{r}_{ij}), \tag{6}$$

where $N_b(i)$ denotes the total number of neighbours of particle i, $m \in [-l, l]$ and $\mathbf{r}_{ij}$ is the center of mass vector defined as $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. The nearest neighbours of particle i are determined according to a cutoff radius which can be determined by the radial distribution functions which are provided in section 4.
The local order parameters are then defined as seen in [14] as

$$q_l(i) = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |q_{lm}(i)|^2} \tag{7}$$

and

$$\omega_l(i) = \frac{\sum\limits_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} q_{lm_1}(i) q_{lm_2}(i) q_{lm_3}(i)}{\left(\sum\limits_{m=-l}^{l} |q_{lm}(i)|^2 \right)^{\frac{3}{2}}}, \tag{8}$$

where $\begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix}$ are the Wigner 3-j symbols and are related to the Clebsch-Gordon Coefficients. This method of crystal structure identification is further improved by introducing the averaged bond order parameters [15]. The averaged form of the local bond order parameter is defined as

$$\bar{q}_l(i) = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |\bar{q}_{lm}(i)|^2} \tag{9}$$

where $\bar{q}_{lm}(i)$ is given by

$$\bar{q}_{lm}(i) = \frac{1}{\tilde{N}_b(i)} \sum_{k=0}^{\tilde{N}_b(i)} q_{lm}(k) \tag{10}$$

It is important to note that the sum runs over all neighbors of particle i as well as particle i itself. Therefore $\bar{q}_l(i)$ contains information of the structure of the first and second shell

around particle i unlike $q_l(i)$ which only contains information of the structure in the first shell around particle i [15].
The averaged order parameter $\bar{\omega}_l$ is defined as

$$\bar{\omega}_l(i) = \frac{\sum\limits_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} \bar{q}_{lm_1}(i)\bar{q}_{lm_2}(i)\bar{q}_{lm_3}(i)}{\left(\sum\limits_{m=-l}^{l} |\bar{q}_{lm}(i)|^2\right)^{\frac{3}{2}}} \tag{11}$$

We shall see later that studying the distribution of local bond order parameters is sufficient for the case of ico-AB$_{13}$ crystal structure identification. Despite this success we employ the assistance of neural networks to perform the same task as neural networks offer additional features.

2.3 Machine Learning

Machine learning can be divided into two main categories, namely supervised and unsupervised learning. Since we wish to classify the particles in a binary mixture as either solid like or fluid like we employ a feedforward neural network (FNN), which classifies as a supervised learning technique [16, 17].
In order to understand how a FNN works we will draw our attention to the most elementary structure in a neural network called an artificial neuron as seen in Figure 2.

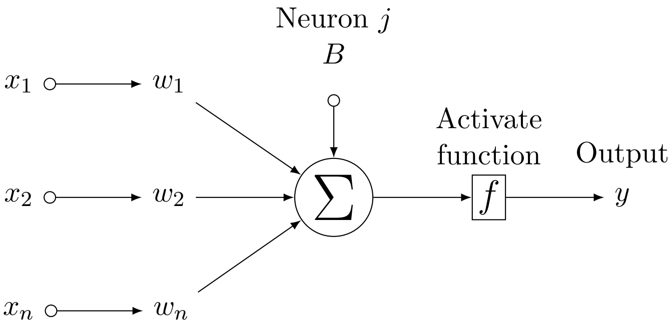


Figure 2: Artificial neuron labeled j with input features $\boldsymbol{X} \in \{x_1, \ldots, x_n\}$, associated weights $\boldsymbol{\omega} \in \{\omega_1, \ldots, \omega_n\}$ and a bias term B.

An artificial neuron labeled j consists of input variables $\boldsymbol{x} \in \{x_1, \ldots, x_n\}$ with corresponding weights $\boldsymbol{\omega} \in \{\omega_1, \ldots, \omega_n\}$ and output $\boldsymbol{y} \in \{y_1, \ldots, y_n\}$. In Figure 2 we simply have a single output, namely y. An artificial neuron can be thought of as a graphical representation of linear regression; where provided with some data $(\boldsymbol{x}, \boldsymbol{y})$ the weights $\boldsymbol{\omega}$ can be tuned in order to describe the trend in the provided data [17]. Predictions can then be made using the tuned weights [17], thus y can be expressed as

$$y = \sum_{i=0}^{n} \omega_i x_i + B \tag{12}$$

Since we are required to determine the weights $\omega_1, \ldots, \omega_n$, supervised learning is subdivided into two main parts: a training and testing phase. We will return to provide a more in depth description of these two phases but for now we focus on the architecture of a FNN. A FNN is simply a collection of artificial neurons to form layers, a general graphical representation of this architecture is given in Figure 3.

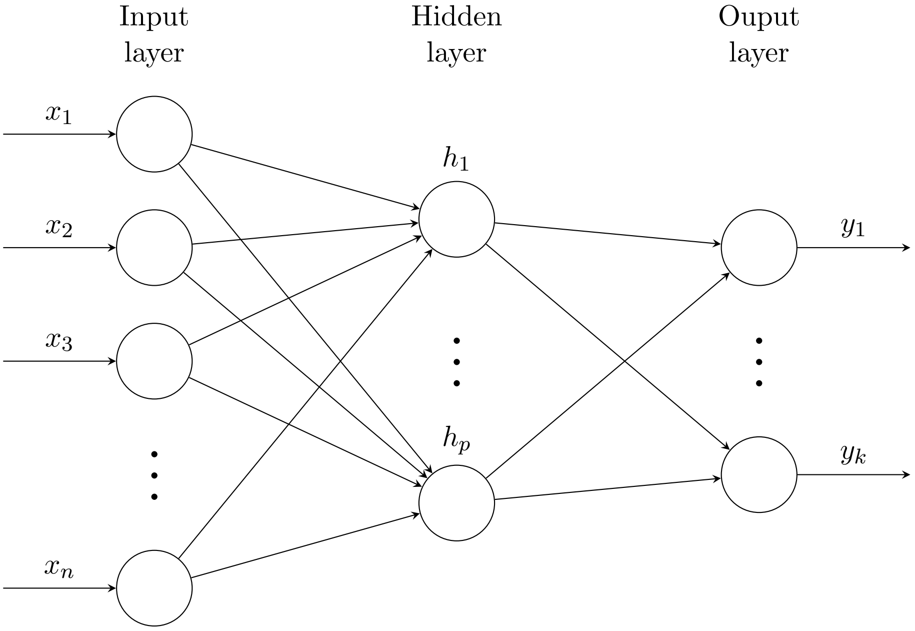


Figure 3: General architecture of a feedforward neural network (FNN) with one hidden layer, input features $\{x_1, \ldots, x_n\}$ and output classes $\{y_1, \ldots, y_k\}$.

As seen in Figure 3, a FNN consists of an input layer which takes as input $\{x_1, \ldots, x_n\}$, called features; a single hidden layer with $\{h_1, \ldots, h_p\}$ neurons. Weights are represented by lines between neurons. Unlike Figure 3 where only one hidden layer is shown one may introduce additional hidden layers in which case the neural network then classifies as a deep neural network [18]. A FNN also consists of an output layer $\{y_1, \ldots, y_k\}$ which are called classes. In the case of classification, these classes are simply called classifiers and are manually chosen by the network architect and not the network itself.

As discussed earlier we revisit the idea of a training and testing phase; however, we shall do so in the context of a FNN.

Training Phase

During the training phase we provide the FNN with what is called a training set $\hat{\boldsymbol{T}}^{\text{train}} \in \{(\hat{\boldsymbol{X}}^{\text{train}} : \hat{\mathbf{Y}}^{\text{train}})\}$ with $\hat{\boldsymbol{X}}^{\text{train}} \in \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_k\}$ and $\hat{\mathbf{Y}}^{\text{train}} \in \{y_1, \dots, y_k\}$ where for each feature set provided $\boldsymbol{x}_i$ a known classifier is given $\mathbf{y}_i$. Using the training set we are then able to determine and tune the weights of the network by minimizing a loss function through a process called back-propagation [16]. There exist numerous optimization techniques used in back-propagation which tune the weights accordingly; however, the correct optimization method depends on the goal of the neural network and the size of training data provided. For the purpose of classification the two most prominent optimizers are stochastic gradient decent, abbreviated as SGD and adaptive moment estimation or Adam as first proposed in [19].

In order to understand how SGD works we need to understand that the loss function of a network is determined by the network's weights. Therefore the loss function forms a surface where each point on the surface corresponds to specific values of the weights and the global minimum of this loss function represents the optimal solution [18, 20]. An example of this surface can be seen in figure 4.

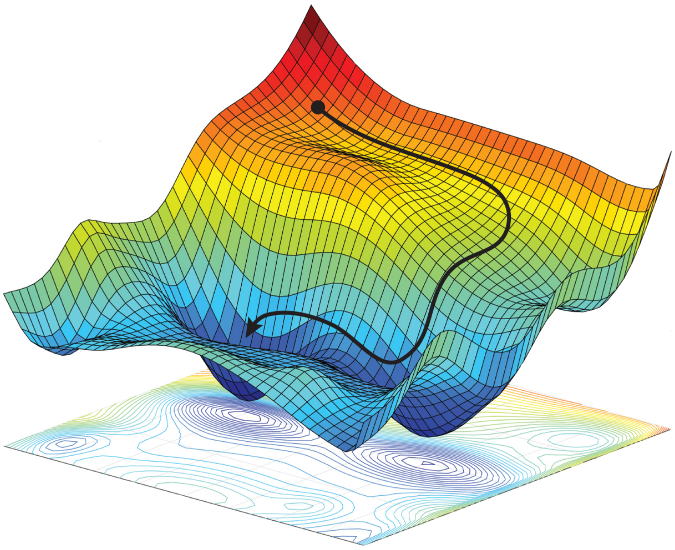


Figure 4: Example of a ”loss surface” where the trajectory represents the stochastic gradient descent to the optimal solution present at the global minimum. Source: [20].

Initially the weights of the network are set to values generated from a uniform distribution, this will correspond to a random position on the surface of the loss function. The slope at this point is then calculated and the weights can thus be readjusted by a step size known as the learning rate so as to decrease the loss. This process is repeated until the loss of the network surpasses some threshold, a trajectory of this process is represented by the black line in figure 4.

If the learning rate is chosen to be too large then it will be to difficult for the algorithm to converge to a possible solution; on the other hand, if the learning rate is selected such that it is too small then convergence to a possible solution will not occur within a reasonable amount of time.

Adam is a more computationally inexpensive based optimizer and is a preferable method to SGD when faced with large data sets [19]. Adam is similar to SGD in the sense that it is a first-order gradient based optimizer [16, 18]; however, the learning rate in Adam is adaptive. We shall not delve too deep into the specifics of Adam but readers are invited to read the paper which first proposed it in [19].

There exist a multitude of different loss functions but for the case of multiclass classification we use the cross entropy loss function given in [16, 17, 18],

$$L(y_{oc}, p_{oc}) = -\sum_{c=1}^{M} y_{oc} \log p_{oc} \tag{13}$$

where y_{oc} is a binary indicator for class label c and observation o that returns 0 if the prediction is incorrect or 1 if the prediction is correct. M represents the total number of classes with $M > 2$, and finally p_{oc} represents the predicted probability that is determined by the network that observation o is of class c.

Testing Phase

During the testing phase we provide the FNN with a test set $\hat{\boldsymbol{T}}^{\text{test}} \in \{(\hat{\boldsymbol{X}}^{\text{test}} : \hat{\boldsymbol{Y}}^{\text{test}})\}$ with $\hat{\boldsymbol{X}}^{\text{test}} \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}$ and $\hat{\boldsymbol{Y}}^{\text{test}} \in \{y_1, \ldots, y_k\}$. The now trained FNN makes predictions $\hat{\boldsymbol{Y}}^{\text{pred}}$ based on $\hat{\boldsymbol{X}}^{\text{test}}$. $\hat{\boldsymbol{Y}}^{\text{pred}}$ is then compared with the correct classifier $\hat{\boldsymbol{Y}}^{\text{test}}$ and the number of correct and incorrect classifications can then be determined. Based on the number of correct classifications an accuracy report of the FNN can be determined.

For reasons given in section 5.2 we resort to the use of a neural network ensemble. A neural network ensemble trains multiple neural networks independently and relies on multiple networks to make predictions/classifications [21]. A neural network ensemble is able to weigh the predictions of some of its constituent networks more than others. Neural network ensembles allow us to reduce the variance in predictions and improve the performance of networks [21, 22].

2.4 Seeding approach to nucleation

We follow a similar seeding approach to nucleation as outlined in [10]. Briefly, this method entails the creation of seeding configurations, whereby a crystalline seed consisting of binary WCA spheres is injected into a binary fluid. It is however important to mention that this method is an approximate one as it relies on the validity of classical nucleation theory, in order to improve the approximation we subject the seeding configurations to an equilibration phase. In this phase the seeding configurations undergo MC simulations at constant environmental conditions. If the radius of the seed is larger than that of the critical radius r_c as mentioned in section 2.1 then the seed will tend to grow in size.

The seeding configurations then undergo MC simulations at various temperatures and pressures which are kept constant during which the number of particles constituting the nucleus is monitored by means of a cluster algorithm.

Once the local structure of a particle has been determined the cluster algorithm then determines whether a particle i is part of a cluster based upon the number of neighboring crystalline particles j, which we shall call the number of bonds. A threshold bond number is defined such that if the number of bonds for a particle i does not exceed this threshold then particle i is disqualified as being part of a crystalline nucleus.

3 Model simulated

We conduct Monte Carlo simulations in the {NPT} ensemble of a binary mixture of soft spheres with N_A species A particles and N_B species B particles with particle diameters σ_A and σ_B respectively. In order to conduct a study of the nucleation of AB_{13} crystals in a binary fluid we need to ensure that the system is present in an environment able to facilitate a phase transformation to the phase of interest. According to [23] and later [8], AB_{13} crystals coexist with a binary fluid at a size ratio between $0.54 \leq \alpha \leq 0.61$, where the size ratio is given by $\alpha = \frac{\sigma_B}{\sigma_A}$, for this reason we choose the size ratio $\alpha = 0.55$. To simulate we use a WCA(12,6) potential which is defined for species $\alpha, \beta \in A, B$.

$$u_{\alpha\beta}^{WCA}(r_{ij}) = \begin{cases} 4\epsilon\left(\left(\frac{\sigma_{\alpha\beta}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{\alpha\beta}}{r_{ij}}\right)^{6}\right) + \epsilon, & \text{if } r_{ij} \leq 2^{\frac{1}{6}}\sigma_{\alpha\beta} \\ 0 & \text{if } r_{ij} > 2^{\frac{1}{6}}\sigma_{\alpha\beta} \end{cases} \tag{14}$$

where r_{ij} is the distance between particle i of species type α and particle j is of species type β and $\sigma_{\alpha\beta}$ represents the effective diameter of particle i and j depending on their respective species. Since $\alpha = 0.55$ we choose $\sigma_A = 1$ as our unit of length. Therefore the effective diameter is $\sigma_{AA} = \sigma_A$ if particle i and j are both species A particles, $\sigma_{BB} = 0.55\sigma_A$ if particle i and j are both species B particles or $\sigma = \frac{\sigma_A+\sigma_B}{2} = 0.775\sigma_A$ if the species of particle i and j are of opposite type. Readers not familiar with Monte Carlo simulations are referred to Appendix A. As Monte Carlo simulations are computationally expensive for large N - the total number of particles; we employ a Verlet Neighborhood List Algorithm which is outlined in further detail in Appendix B.

4 System properties

4.1 Equations of state

In this section we show the results obtained after performing MC simulations for crystalline and fluid configurations in an {NPT} ensemble for temperatures $k_BT/\epsilon = 0.1$, $k_BT/\epsilon = 0.2$ and $k_BT/\epsilon = 0.025$. An AB_{13} configuration in the crystalline phase was generated for $N = 896$ particles at a density $\rho\sigma_L^3 = 4.5$. This configuration was used in three separate MC simulations at temperatures $k_BT/\epsilon = 0.1$, $k_BT/\epsilon = 0.2$ and $k_BT/\epsilon = 0.025$, which were kept constant in each case. In each simulation the pressure was set at $\beta P\sigma_L^3 = 70.0$ and kept constant while the density $\rho\sigma_L^3$ was monitored. Once the density equilibrated around some average value the average density would then be recorded, subsequently the pressure was

then lowered. This process was repeated steadily reducing the pressure until the pressure $\beta P\sigma_L^3 = 5.0$. The results obtained for the temperature $k_BT/\epsilon = 0.025$ are given in Figure 5 represented by the *red* curve.
A similar procedure was repeated for a fluid with a composition $N_A/(N_A + N_B) = 1/14$ that was prepared at a low density $\rho\sigma_L^3 = 0.5$. This configuration was also used in three separate MC simulations at temperatures $k_BT/\epsilon = 0.1$, $k_BT/\epsilon = 0.2$ and $k_BT/\epsilon = 0.025$; however, in each case the temperature was initially set at $\beta P\sigma_L^3 = 5.0$ and was steadily increased while monitoring the average density $\rho\sigma_L^3$ at regular intervals. The fluid curve calculated for the temperature $k_BT/\epsilon = 0.025$ is given in figure 5 and is represented by the *blue* curve.

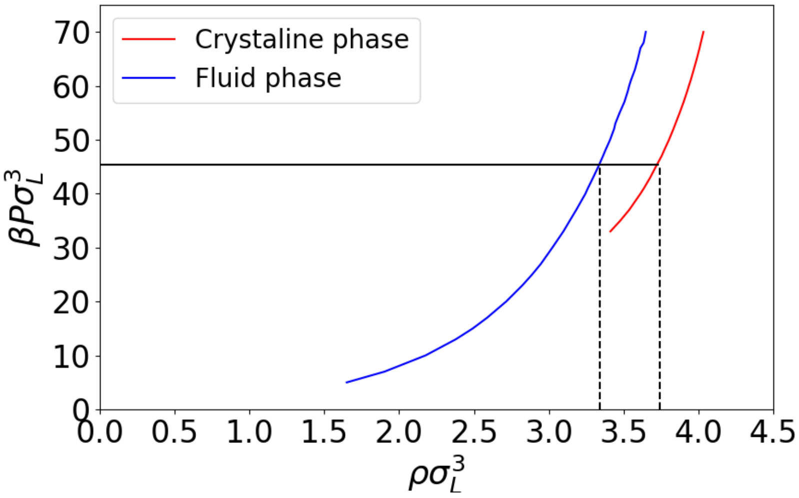


Figure 5: Equations of state obtained for temperature $k_BT/\epsilon = 0.025$ displayed as pressure ($\beta P\sigma_L^3$) as a function of the number density ($\rho\sigma_L^3$). The blue curve represents the fluid phase while the red represents the crystalline phase.

Figure 5 displays two main curves. The curve that exists at a high density represents the crystalline phase (red curve) while the curve that exists at low density represents the fluid phase (blue curve). Hence the two curves are referred to as the crystalline and fluid curves respectively. In Figure 5 we can see that the crystalline curve displays a rapid decrease in density in the pressure region $30.0 < \beta P\sigma_L^3 < 33.0$. Similar jumps in density are observed for temperature $k_BT/\epsilon = 0.1$ in the pressure region $33.0 < \beta P\sigma_L^3 < 35.0$ and for the temperature $k_BT/\epsilon = 0.2$ in the pressure region $35.0 < \beta P\sigma_L^3 < 37.0$. These jumps in densities are attributed to the AB_{13} crystalline phases melting into the fluid phase. At higher temperatures $k_BT/\epsilon = 0.1$ and $k_BT/\epsilon = 0.2$ it is observed that the AB_{13} crystalline phases melt at higher pressures.
From the EOS the regimes at which coexistence occurs could be estimated. For a temperature $k_BT/\epsilon = 0.025$ we find coexistence at $\rho\sigma_L^3 > 3.41$ and $\beta P\sigma_L^3 > 33.0$, for $k_BT/\epsilon = 0.1$

we find coexistence at $\rho\sigma_L^3 > 3.60$ and $\beta P\sigma_L^3 > 35.0$ and finally for $k_B T/\epsilon = 0.2$ we find coexistence at $\rho\sigma_L^3 > 3.76$ and $\beta P\sigma_L^3 > 37.0$.

4.2 Coexistence pressure

In the following subsection we present the results obtained in order to determine the coexistence pressure $\beta P_{coex}\sigma_L^3$, the pressure where the coexistence of the crystalline AB$_{13}$ phase and the fluid phase coexist. Using thermodynamic integration the free energy of the AB$_{13}$ crystalline phase and the binary fluid phase was calculated. Using a common tangent construction to the free-energy curves we find coexistence at a pressure $\beta P_{coex}\sigma_L^3 = 45.38$ as shown in figure 6. To confirm coexistence we also compute the chemical potential $\beta\mu$ using the Helmholtz free energy, and plot the chemical potential $\beta\mu$ versus the pressure $\beta P\sigma_L^3$ for both the crystalline AB$_{13}$ and fluid phase as shown in figure 7.

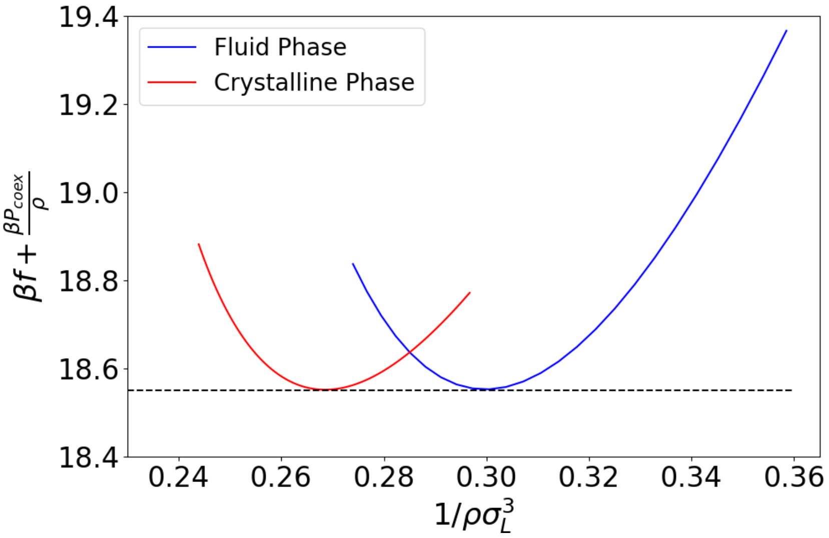


Figure 6: Free energy density $f = F/V$ as a function of the number density ρ for the AB$_{13}$ crystalline phase (*red*) as well as the fluid phase (*blue*).

It is a well known fact that at the coexistence point the two phases in question must have equal temperatures $T_F = T_C$, equal chemical potentials $\mu_F = \mu_C$ as well as equal pressures $P_F = P_C$; where the subscript C represents the crystalline phase and the subscript F represent the fluid phase. Therefore the point of intersection of the two lines in figure 7 represents the point of coexistence as this point satisfies all three conditions. It is found again that the coexistence pressure at this point (denoted by the dashed black line in figure 7) occurs at $\beta P_{coex}\sigma_L^3 = 45.38$.

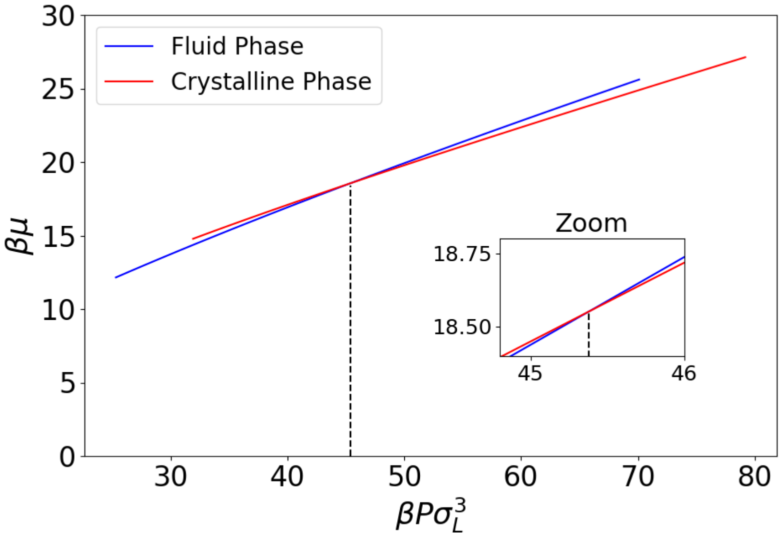


Figure 7: Chemical potential $\beta\mu$ as a function of the pressure $\beta P\sigma_L^3$ for the AB$_{13}$ crystalline phase (*red*) as well as the fluid phase (*blue*).

4.3 Radial distribution functions

The radial distribution functions or pair correlation functions describe the density variation of particles as a function of the radial distance from a particle. An investigation of the radial distribution of particles within the crystalline AB$_{13}$ phase was calculated, showing the correlation between the species A and species A particles ($g_{LL}(r/\sigma_L)$), between the species B and species B particles ($g_{SS}(r/\sigma_L)$) and the correlation between the species A and species B particles ($g_{SL}(r/\sigma_L)$). The results are presented in figure 8.

The pair correlation functions were calculated by dividing the radial distance from a particle i into bins of a certain size given by $\Delta r_{bin} = 0.5L/\text{Number of bins}$, where L represents the length of the box. Where the number of bins was chosen such that the resulting pair correlation functions became smooth. Each bin then creates a shell around the particle i with a thickness Δr_{bin}, the number of particles within each shell is then calculated. This process was repeated for a number of particles i and then an average was taken over the number of particles considered. The species of particle i and the species of particles counted within each shell depends on the pair correlation function under investigation.

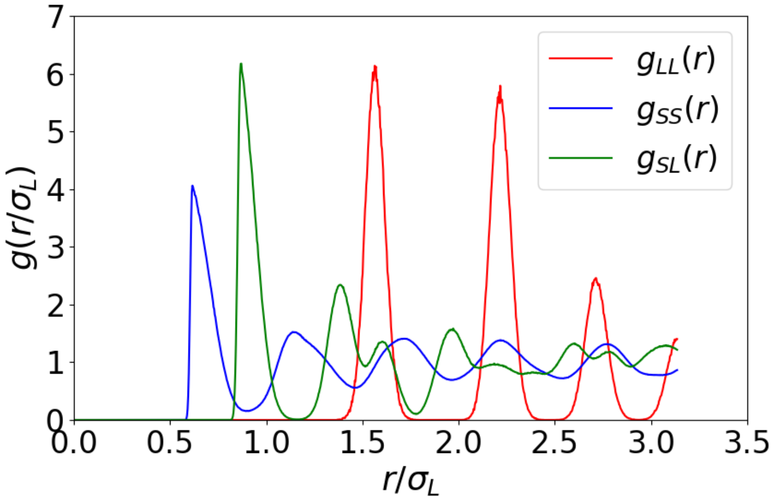


Figure 8: Pair correlation functions $g_{LL}(r/\sigma_L)$ which shows the correlation between species A and species A particles, $g_{SS}(r/\sigma_L)$ which shows the correlation between species B and species B particles, and finally $g_{SL}(r/\sigma_L)$ which shows the correlation between species B and species A particles at a temperature $k_BT/\epsilon = 0.025$ and a pressure $\beta P\sigma_L^3 = 40.0$.

With the radial distribution functions given in Figure 8 we are able to determine the cutoff radius necessary to determine the first shell of neighbors N_b for a particle i. The first peak encountered within the radial distribution function represents the first shell of nearest neighbors around particle i, thus the first well after this peak represents an ideal cutoff point to define nearest neighbors. Therefore, using the results obtained in figure 8 the cutoff radius necessary to find the first shell of species A neighbors to a species A particle is given as a point in the first well of $g_{LL}(r/\sigma_L)$, thus the cutoff radius is $r_c^{LL}/\sigma_L = 1.75$. Similarly the pair correlation between species B and species B particles gave the cutoff radius for the first shell of species B particles around species B particles as $r_c^{SS}/\sigma_L = 0.8$, taken from $g_{SS}(r/\sigma_L)$. Finally the cutoff radius which defines the first shell of species A neighbors around a species B particles as well as the cutoff radius which defines the first shell of neighbors of species B particles around a species A particle is taken from the pair correlation function $g_{SL}(r/\sigma_L)$ and is given by $r_c^{SL}/\sigma_L = 1.25$.

In addition, we measure the radial distribution functions for $g_L(r/\sigma_L)$ which shows the correlation between species A and all species particles, $g_S(r/\sigma_L)$ which shows the correlation between species B and all species particles, and finally $g(r/\sigma_L)$ which shows the correlation between particles irrespective of the species we show the result in figure 9 and find that these radial distribution functions are characteristic of a dense fluid.

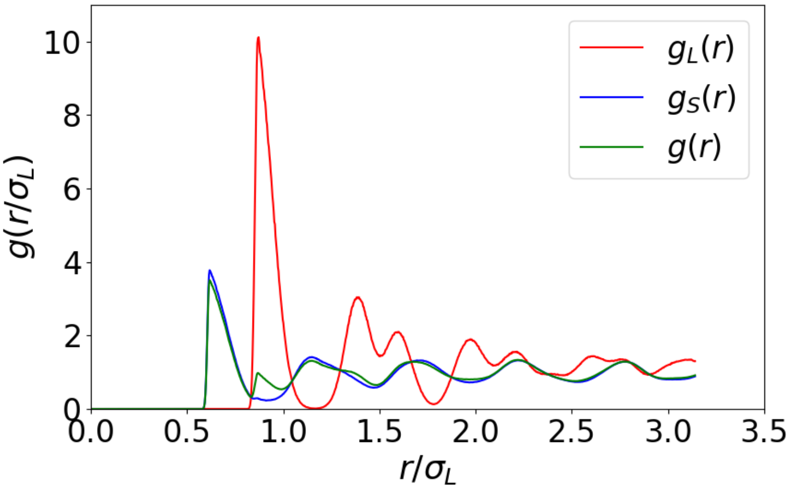


Figure 9: Pair correlation functions $g_L(r/\sigma_L)$ which shows the correlation between species A and all species particles, $g_S(r/\sigma_L)$ which shows the correlation between species B and all species particles, and finally $g(r/\sigma_L)$ which shows the correlation between two particles irrespective of the species at a temperature $k_\beta T/\epsilon = 0.025$ and a pressure $\beta P\sigma_L^3 = 40.0$.

5 Phase identification

5.1 Bond order parameters

Now that we have determined the respective cutoff radius for species A neighbors, species B neighbors and general particle neighbors as seen in section 4, we are now able to determine the nearest neighbors N_b necessary to calculate the order parameters defined in subsection 2.2. In this section we focus on the averaged bond order parameters for the reason that they are preferable to the non averaged bond order parameters as seen in [15]. The averaged bond order parameters for approximately 200 crystalline species A particles, 200 crystalline species B particles of an AB_{13} crystal, 200 particles of a pure FCC of large particles and 200 fluid particles for a composition of $N_L/(N_L + N_S) = 1/14$ were calculated. These particles were selected from configurations prepared in a MC simulation at a temperature $k_B T/\epsilon = 0.025$ and pressure $\beta P\sigma_L^3 = 40.0$. The averaged bond order parameters that were considered were $\bar{q}_l(i)$ and $\bar{\omega}_{l'}(i)$ where $l \in \{1, 8\}$ and $l' \in \{2, 4, 6, 8\}$. Of these bond order parameters that were calculated the most promising were found to be $\bar{q}_4$, $\bar{q}_6$, $\bar{q}_8$, $\bar{\omega}_4$ and $\bar{\omega}_6$. Firstly we display the probability distributions for the $\bar{q}_4$ and $\bar{q}_6$ bond order parameters for each phase labeled accordingly, as seen in the figure below 10.

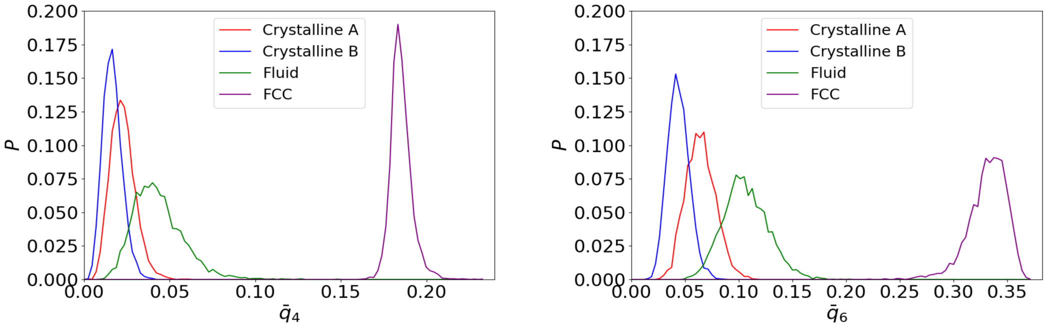
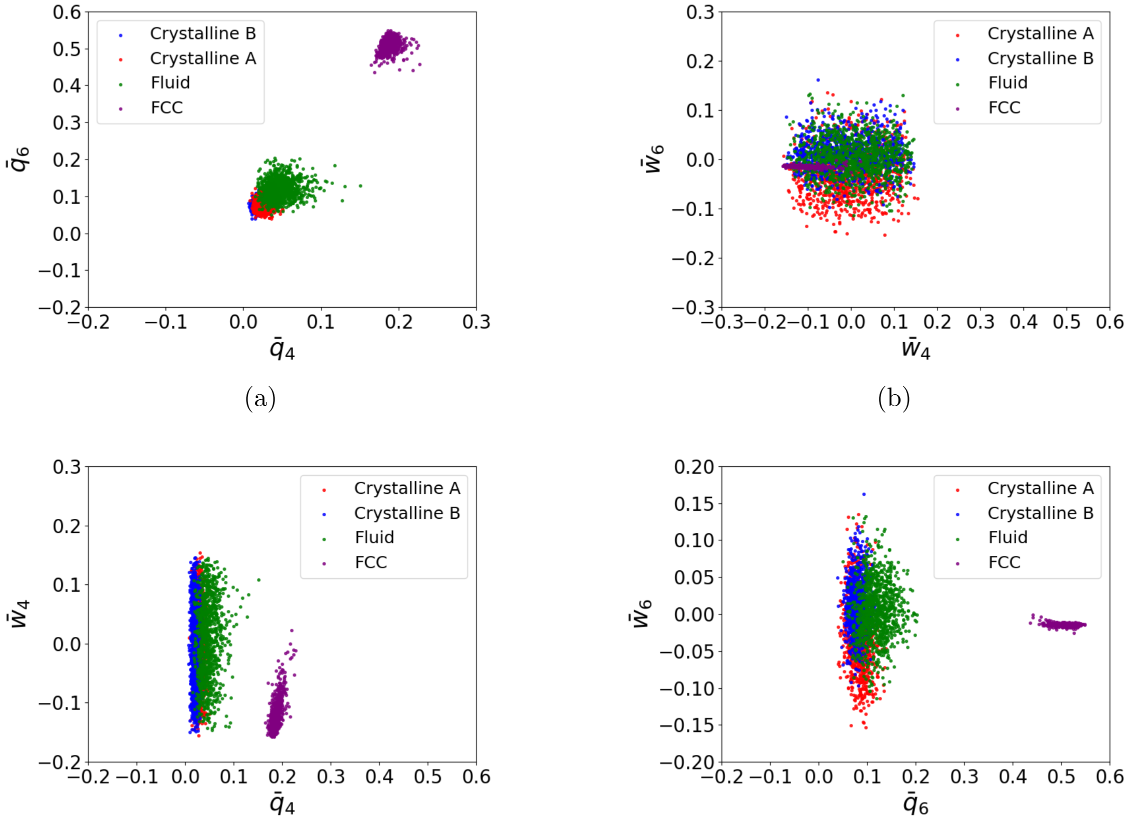


Figure 10: The following figure shows the probability distributions for the local order parameters, (*left*) the $\bar{q}_4$ parameter and (*right*) the $\bar{q}_6$ parameter for the crystalline species A particles (*red*), crystalline species B particles (*blue*), fluid phase (*green*) and the FCC phase (*pink*) at a temperature $k_BT/\epsilon = 0.025$ and pressure $\beta P\sigma_L^3 = 40.0$.

From the distributions in figure 10 we see that the distribution of the bond order parameters $\bar{q}_4$ and $\bar{q}_6$ for the FCC phase is easily distinguishable from the other phases due to the spread of the possible values. Unfortunately the distribution of the bond order parameters $\bar{q}_4$ and $\bar{q}_6$ for the crystalline species A and B are not easily distinguishable from one another or from the fluid phase due to the observed spread of the possible values. Since the individual parameters were not sufficient to distinguish the phases from one another the bond order parameter planes were then considered.

More specifically we studied the $q_n - q_m$, $\bar{q}_n - \bar{q}_m$, $\omega_n - \omega_m$ and $\bar{\omega}_n - \bar{\omega}_m$, $\omega_n - q_m$ and $\bar{\omega}_n - \bar{q}_m$ planes, with $m, n \in \{2, 4, 6, 8\}$ and $m \neq n$. These planes were constructed by plotting the bond order parameters as a set of points on a two dimensional plane. The most promising of these planes studied were the $q_6 - q_4$, $\bar{q}_6 - \bar{q}_4$, $\omega_6 - \omega_4$ and $\bar{\omega}_6 - \bar{\omega}_4$ planes which have been provided in figure 11 below.



(a)

(b)

(c)

(d)

Figure 11: Scatter plots of the bond order parameter values of a binary fluid phase with composition $N_L/N_L + N_S = 1/14$, an AB$_{13}$ crystalline phase and an FCC crystalline phase in (a) the $\bar{q}_6 - \bar{q}_4$ plane, in (b) the $\bar{\omega}_6 - \bar{\omega}_4$ plane, (c) the $\bar{\omega}_4 - \bar{q}_4$ plane and in (d) the $\bar{\omega}_6 - \bar{q}_6$ plane at a temperature $k_\beta T/\epsilon = 0.025$ and pressure $\beta P \sigma_L^3 = 40.0$.

We see from the planes displayed in figure 11 that we obtain similar results to that observed in figure 10 in that we are able to differentiate the FCC phase from the other three phases. We can see this by observing that the distribution of the FCC points in the $\bar{q}_6 - \bar{q}_4$, $\bar{\omega}_4 - \bar{q}_4$ and $\bar{\omega}_6 - \bar{q}_6$ planes do not overlap with the distribution of points of the other phases. However, the distribution of the crystalline species A points, crystalline species B points and the fluid points all overlap with one another and so these distributions are unreliable in being able to differentiate the different phases from one another.
Fortunately from the $\bar{\omega}_8 - \bar{\omega}_6$ plane in the three dimensional $\bar{q}_8 - \bar{\omega}_6 - \bar{\omega}_8$ space in figure 12, we can distinguish the difference between the crystalline species A and crystalline species B particles from the FCC particles and the fluid phase particles. This is due to the fact the distribution of possible points do not overlap with one another.

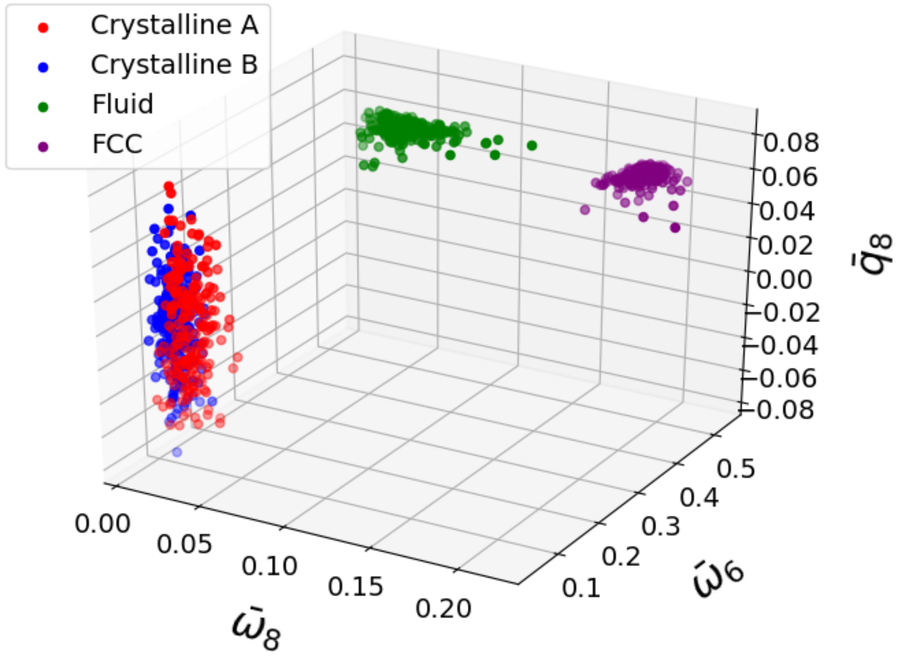


Figure 12: This figure shows the $\bar{q}_8 - \bar{\omega}_6 - \bar{\omega}_8$ surface for the crystalline A (*red*), crystalline B (*blue*), fluid (*green*) and FCC phase .

But all is not lost, since we have the particle diameters $\sigma(i)$ readily available to us, where $\sigma(i) = \sigma_L$ if particle i is of species type A and $\sigma(i) = \sigma_S$ if particle i is of species type B, then we can easily distinguish the four phases from one another using figure 12 in conjunction with the diameter of the particles. We now move to the use of neural networks as they offer a means to define an explicit function[ii] which classifies the local structure of a particle and is thus a more elegant means of local structure identification.

[ii] We say function as a neural network is essentially a function that has been fitted according to training data.

5.2 Neural networks

In this section we enlist the assistance of neural networks[iii] to make predictions of the local structure of a particle i. Before we could do this we constructed a data set through which we could define a training set and a test set, this is presented in subsection 5.2.1. Then based on this data set, neural network architectures were tested and evaluated in order to determine the best network possible for the task of local structural identification, which is discussed in detail in subsection 5.2.2. It was then found that a neural network ensemble was preferable to a singular network and the results of which are presented in section 5.2.3.

5.2.1 Preprocessing

In the following section we outline the features that we chose for our neural networks to train as well as to use neural networks to make classifications. The number of features as well as the number of classes dictates the the number of input and output nodes in our neural networks. Firstly we constructed a data set that could then be used for training as well as testing the neural networks.

Following the recipe in [24] we constructed a vector $\boldsymbol{Q}(i)$ for each particle i, this vector consists of a total of 25 features. 16 of these features consisted of the averaged bond order parameters $\bar{q}_l(i)$ and $\bar{q}_l^{LL(SS)}(i)$ where $l \in \{1, 8\}$. 8 of the features are given by $\bar{\omega}_{l'}(i)$ and $\bar{\omega}_{l'}^{LL(SS)}(i)$ where $l' \in \{2, 4, 6, 8\}$. Where the superscript $LL(SS)$ denotes parameters calculated from only considering the large (small) nearest neighbors of large (small) particles. The last feature is simply the relative diameter of colloidal particle i given by $\sigma^*(i) = \sigma(i)/\sigma_L$ where $\sigma(i) = \sigma_L$ if particle i is of species type A, or $\sigma(i) = \sigma_S$ if particle i is of species type B. [24]. This input vector can be expressed as

Table 1: Labeling assignment for each respective class.

Label:	Class (Phase):
0	Crystalline species A
1	Crystalline species B
2	Fluid
3	FCC

$$\boldsymbol{Q}(i) = (\{\bar{q}_l(i)\}, \{\bar{\omega}_{l'}(i)\}, \{\bar{q}_l^{LL(SS)}(i)\}, \{\bar{\omega}_{l'}^{LL(SS)}(i)\}, \sigma^*(i)) \tag{15}$$

Using this input vector we constructed a main data set consisting of 627200 ico-AB$_{13}$ crystalline particles and 627200 AB$_{13}$ fluid particles, prepared at a temperature $k_BT/\epsilon = 0.025$ and pressure $\beta P\sigma_L^3 = 45.38$. Additionally, $\boldsymbol{Q}(i)$ was determined for 200000 species A and B particles within the FCC phase [iv].

[iii] Python offers a compilation of machine learning libraries including SciKit Learn, Keras, TensorFlow and PyTorch. Readers not familiar with machine learning packages in Python but would like to gain a working knowledge are recommended to start with Keras and SciKit Learn.

[iv] SciKit Learn offers unique and efficient pre-processing functions that prepare data for NN use.

These particles were labeled according to the classes in Table 1 [v], therefore our neural network architecture will have 4 output nodes.
From the data set we then separated the set such that 75% of the set formed part of the training set while the remaining 25% was then selected as part of the testing set. Before selecting configurations from the main data set to form part of the training set, the main data set was shuffled and configurations were selected randomly. Before the data was split into the training and testing sets the data was normalized. Normalizing data for the network is vitally important as some features may be weighted higher than other features purely because of the difference in magnitude between the two features. So in order to eliminate this issue we normalized the data used to test and train the network.

5.2.2 Network architectures

Based on the number of features that were selected the number of nodes within the input layer was set to 25 for all neural networks tested; additionally, due to the number of classes selected all neural networks were set to have 4 output nodes within the output layer.

Neural networks with one hidden layer were trained each with various number of nodes in the hidden layer, ranging between 4 to 25 nodes - the same number of nodes within the input layer. The accuracy of these networks were then compared with one another and it was found that networks with more than 10 nodes within the hidden layer converged to accuracies greater than 90%[vi]. Therefore neural networks with 25 nodes in the hidden layer were selected as the default network architecture as this would be more than sufficient to obtain an accurate neural network. We then trained two neural networks that both had this network architecture but with different optimizers. The first network trained was trained using SGD with a learning rate = 0.001, the second network was then trained using the Adam optimizer. The overall performance of these two networks is abbreviated in table 2 below.

Table 2: Overall performance of neural networks trained for each optimizer.

Neural Network	Optimizer	Network Accuracy (%)
1	SGD	99.88
2	Adam	99.78

The overall accuracy of the two networks is not indicative of whether the networks actually perform in the way that we want them to. Therefore it was necessary to conduct a further analysis into the training and testing loss and accuracy of the two networks. During the training phase the loss is measured in order to make the appropriate changes to the

[v] This type of labeling is necessary to construct the target vectors which label the classes in Keras and TensorFlow.

[vi] Another test which allows one to compare networks on set data is called the cross validation test.

weights of the network during back propagation; however, during the testing phase the loss is measured to evaluate the performance of the network after the training phase. These results are presented in figures 13 and 14. The training and testing phases are repeated over epochs, so each epoch represents a training phase followed by a testing phase. Each epoch that is not the initial epoch uses the network that was trained and tested in the previous epoch. In this way a neural network was developed.

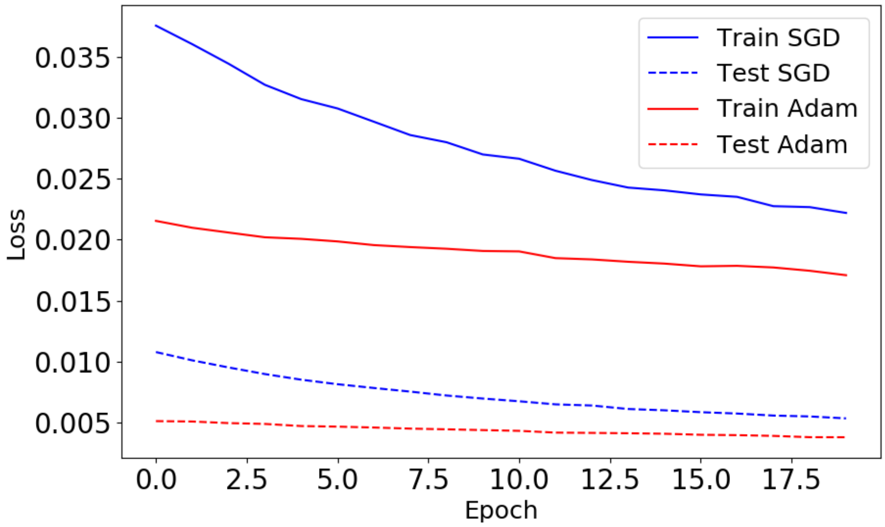


Figure 13: The training and testing loss functions versus epoch obtained for the training and testing data sets using the SGD optimizer and the Adam optimizer, for the neural network.

From the loss curves seen in figure 13 we are able to deduce that the Adam optimization method allows us to converge to a solution a lot more rapidly when compared to the SGD method. Therefore Adam is an ideal choice for an optimizer when data is scarce. We notice that in both cases that the loss for the test set is lower than that of the training set, this is expected as the loss of the training set is the loss over the entire set, whereas the loss of the test set is only after the test has been performed.

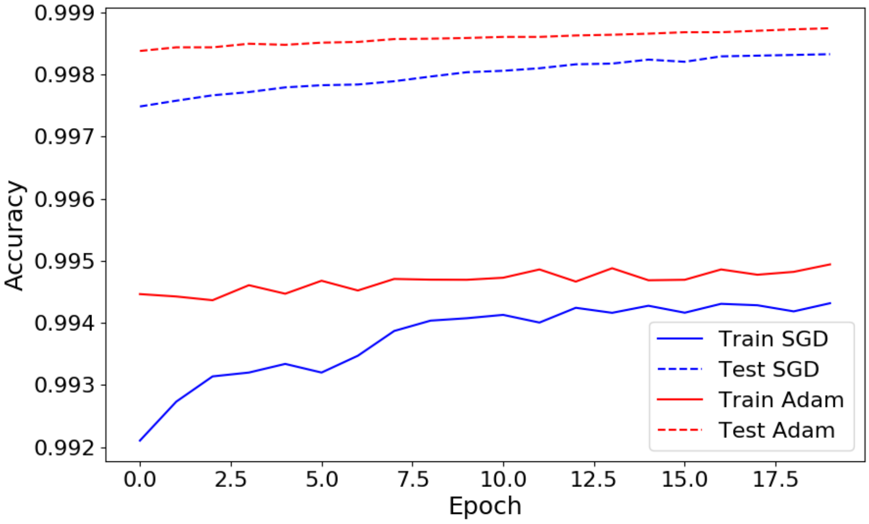


Figure 14: The accuracy as a function of epoch obtained for the training and testing data sets using the SGD optimizer and the Adam optimizer for the neural network.

From figure 14 we observe the results of the accuracy curves for the two networks, which reaffirms the results observed in figure 13; whereby, we find that networks trained using the Adam optimizer converge more rapidly towards a solution.
But a visual confirmation of these networks was necessary, these trained networks were then tested on a seeding configuration consisting of an ico-AB_{13} crystalline seed surrounded by a fluid. Particles were then colour coded according to the local structure as predicted by the neural networks using the Adam and SGD optimizer. Cross sections of these configurations are given in figure 15 below. As seen in figure 15, both networks present the majority of mislabels within the fluid phase - whereby particles that belong to the fluid phase are labeled as either crystalline species A or B, or the crystalline FCC phase. Both networks seem to present the same level of success in identifying the local structure of particles within the fluid phases; however, we notice that in (a) large crystalline species A particles have been mislabeled in the seed as fluid phase particles. On the other hand this is not the case in (b). The possible reason for this discrepancy is discussed further in section 7.

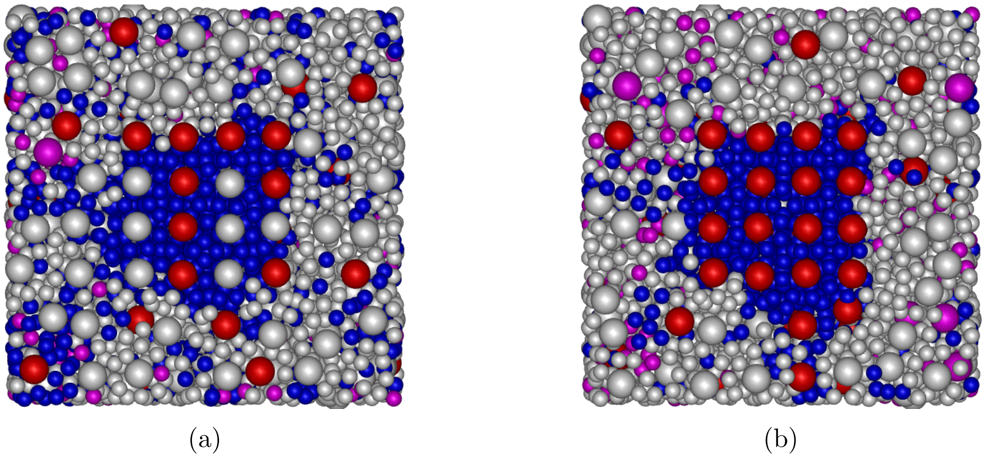


Figure 15: The following figure shows a cross section of the seeding configuration with the respective predictions using the Adam optimizer (a) and SGD optimizer (b). Labels are in accordance with Table 1: crystalline species A - Red, crystalline species B - Blue, Fluid - grey and FCC - pink.

5.2.3 Neural network ensemble

In an attempt to improve the accuracy of the networks we could reduce the variance of the classifications through the development of a neural network ensemble [17, 16]. A neural network ensemble is a network which consists of multiple feed forward neural networks trained separately. When a prediction is made using a neural network ensemble each feed forward neural network with in the ensemble makes a separate prediction on the local structure of a particle, called a vote; the neural network ensemble then counts the votes of each neural network and classifies the local structure according to the majority vote. We constructed a neural network ensemble using 4 feed forward neural networks, each network with in this ensemble contained one hidden layer with each feed forward neural network consisting of various number of nodes, the 4 feed forward neural network's architecture and performance are summarized in table 3.

Table 3: Performance and number of nodes with in the first hidden layer of the 4 feed forward neural networks which constitute the neural network ensemble.

Neural Network	Nodes in hidden layer	Network Accuracy (%)
3	14	99.86
4	16	99.88
5	24	100
6	24	99.98

The neural network ensemble was constructed such that all neural networks possessed an equal vote weighting. The neural network ensemble was then tested on the seeding configuration and its predictions are visualized in figure 16. When the visualization of figure 16 is compared with that of 15 (b) we see that the neural network ensemble has less misclassifications of the local structure of particles contained within the fluid than that of the forward neural network trained using the SGD optimizer.

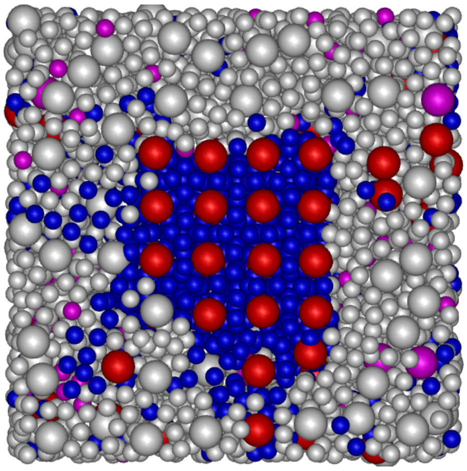


Figure 16: Cross section of the seeding configuration with the respective predictions using the neural network ensemble.

Although neural networks allow us to make accurate classifications of the local structure of a colloidal particle based upon the features that we train the neural networks on, in this case the bond order parameters; they also offer information on what features in particular are mainly used in order to make a classification. This warrants a probe into the trained neural networks trained and so we investigate feature importance.

5.2.4 Feature importance

Feature importance was then determined using neural network 2 and 3 as summarized in table 3. Feature importance measures a neural networks dependence on a feature in order to make a classification. In order to calculate the importance of a feature in a neural network, the network makes predictions using the exact same data used to train the network; however, the data representing the feature being tested is randomly shuffled. The reason why the feature set is shuffled is to ensure that the feature contains incorrect data corresponding to a particular classification. Therefore if the neural network relies heavily on a particular feature it's accuracy would decrease when the feature is shuffled; however, if the neural network does not rely heavily on a particular feature, the overall accuracy of the network would not decrease as much. This method allows us to compare the importance of certain features to a particular trained network. Thus the accuracy of the network was measured and the difference $Acc_{Network} - Acc_{Feature}$ was calculated, where $Acc_{Network}$ is the accuracy of the network and $Acc_{Feature}$ is the accuracy of the network when a features data has been shuffled. Both network 2 and 3 presented the same feature importance and so we only provide the feature importance of network 2 as outlined in figure 17.

We can not say anything about the importance of a feature based on the accuracy difference itself; however, we may draw conclusions when we compare the accuracy differences between the features, which is precisely the case in figure 17. And so we see that both neural networks rely mostly on $\bar{\omega}_8$ to make predictions while not relying on $\hat{q}_1$ at all (note that $\hat{q}_1 = \bar{q}^{LL(SS)}$, this new notation was employed to simplify figure 17). In fact $\{\hat{q}_1, \hat{q}_2, \hat{q}_3, \hat{q}_4, \hat{q}_5, \hat{q}_6, \hat{q}_7\}$ are seen to have the least amount of predictive power within the network. All other bond order parameters namely $\hat{q}_8$, $\{\hat{\omega}_2, \hat{\omega}_4, \hat{\omega}_6, \hat{\omega}_8\}$, $\{\bar{q}_1, \bar{q}_2, \bar{q}_3, \bar{q}_4, \bar{q}_5, \bar{q}_6, \bar{q}_7\, \bar{q}_8\}$ and $\{\bar{\omega}_2, \bar{\omega}_4, \bar{\omega}_6, \bar{\omega}_8\}$ listed in order of importance. We do not include σ^* in this evaluation as we are only interested in the affect of the bond order parameters on the network.

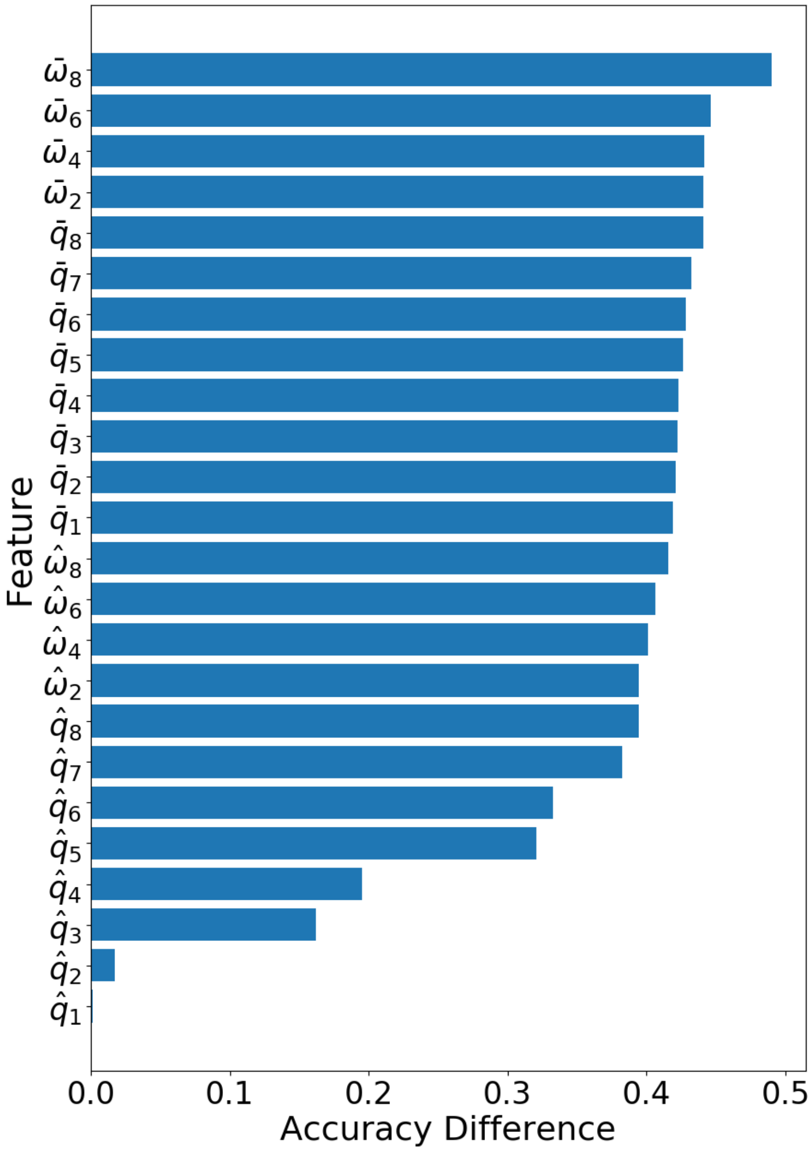


Figure 17: Feature importance (Network 2) measured as the accuracy difference $Acc_{Network} - Acc_{Feature}$, where $Acc_{Network}$ is the accuracy of the network and $Acc_{Feature}$ is the accuracy of the network with the respective feature shuffled.

6 Nucleation

Now that we are more than capable of determining the local structure of a particle either using the $\bar{\omega}_8 - \bar{\omega}_6$ plane together with knowledge of the diameter of the particle $\sigma(i)$ or using the neural network ensemble developed in section 5.2. In this section we opted to use the neural network ensemble as the network had a means of performing predictions that was readily available. Four seeding configurations were then created with various seed sizes in various system sizes, the exact configurations are summarized in table 4.

Table 4: Seeding configurations XL, L, M and S measured in terms of unit cells inserted into a system with $N \times 10^3$ particles within the fluid phase.

Seed	N ($\times 10^3$)	Seed Size (unit cells)
XL	60	27 (3054 particles)
L	40	27 (3054 particles)
M	25	8 (896 particles)
S	15	8 (896 particles)

We measure the seed size in terms of unit cells, which in this case a single unit cell consists of 56 particles in total. 4 of these particles are species A particles arranged in a cubic structure while the remaining 52 particles are species B particles and are arranged in half a icosahedral structure.

A unit cell can be visualized in figure 18 where the cubic structure of the species A particles can be clearly seen. These seeding configurations were then subjected to MC simulations under constant pressure $\beta P \sigma_L^3 = 62.00$ well above the coexistence pressure $\beta P_{coex} \sigma_L^3 = 45.38$ and temperature $k_B T / \epsilon = 0.025$ and allowed to equilibrate.
As we see from Figure 18 the seeds have an overall cubic structure which is not consistent with classical nucleation theory as it assumes that the nucleus has a spherical structure, it is for this reason that we equilibrate the system. This ensures that the nucleus becomes "smooth" - more spherical.

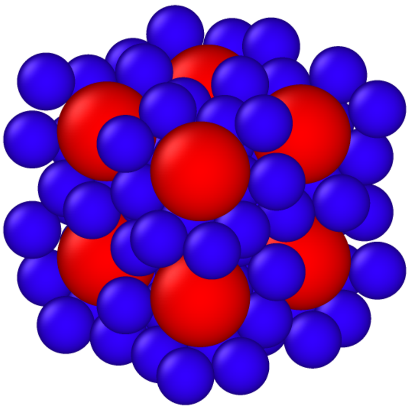


Figure 18: A single unit cell.

The bond order parameters are then calculated for each configuration generated at each MC cycle and was then fed to the neural network ensemble constructed in section 5.2.3; consequently, the local structure of each particle was then classified. Once these classifications were made a clustering algorithm then analyzed each configuration outlining the clusters present as well as monitor their sizes as seen in the examples provided in figure 19 below.

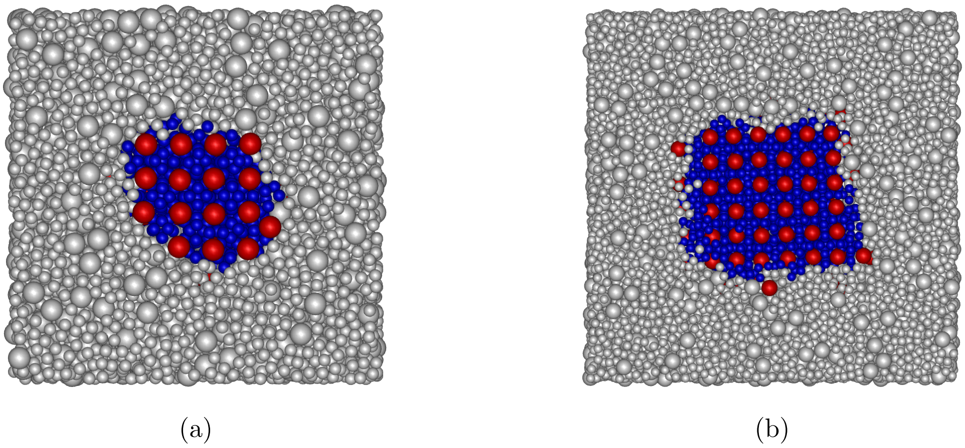
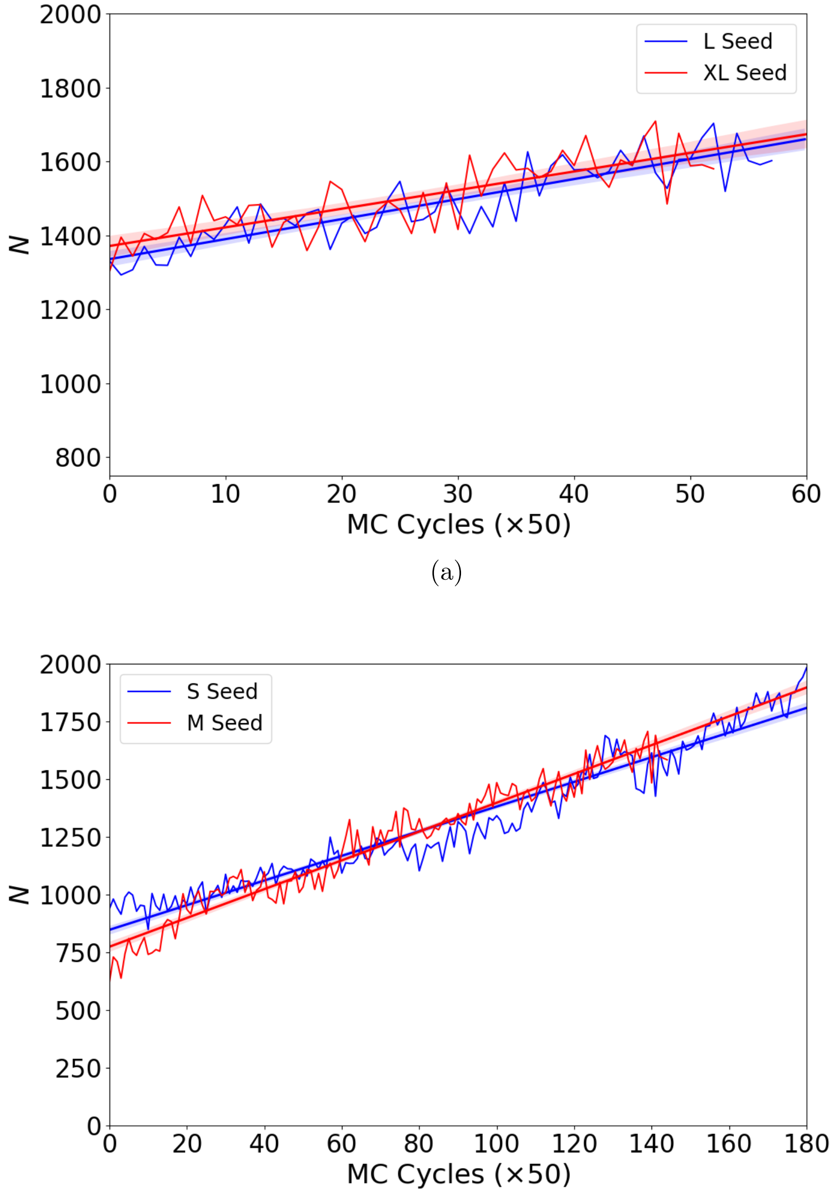


Figure 19: The following figure shows a cross section of the seeding configuration after the cluster algorithm has been applied. (a) Small seeding configuration and (b) Large seeding configuration. Labels are in accordance with Table 1: crystalline species A - Red, crystalline species B - Blue, Fluid - grey and FCC - pink.

In figure 20a we display the nucleus size as a function of every 50th MC cycle for the configurations which contained the large seeding configuration (L) and the extra large seeding configuration (XL), as labeled in table 4; while in figure 20b we display the results for the case of the small seeding configuration (S) and the medium seeding configuration (M). We found that the nucleus size displayed an overall increase in size as a function of MC cycles in all seeding configurations. This trend is highlighted with the linear trend fitted to the trajectories in figures 20a and 20b. Since the XL, L, M and S seeding configurations present results which demonstrate that growth of the seed is taking place it could be concluded that the seed radius in each case exceeded that of the critical nucleation radius r_c as mentioned in section 2.



(a)

(b)

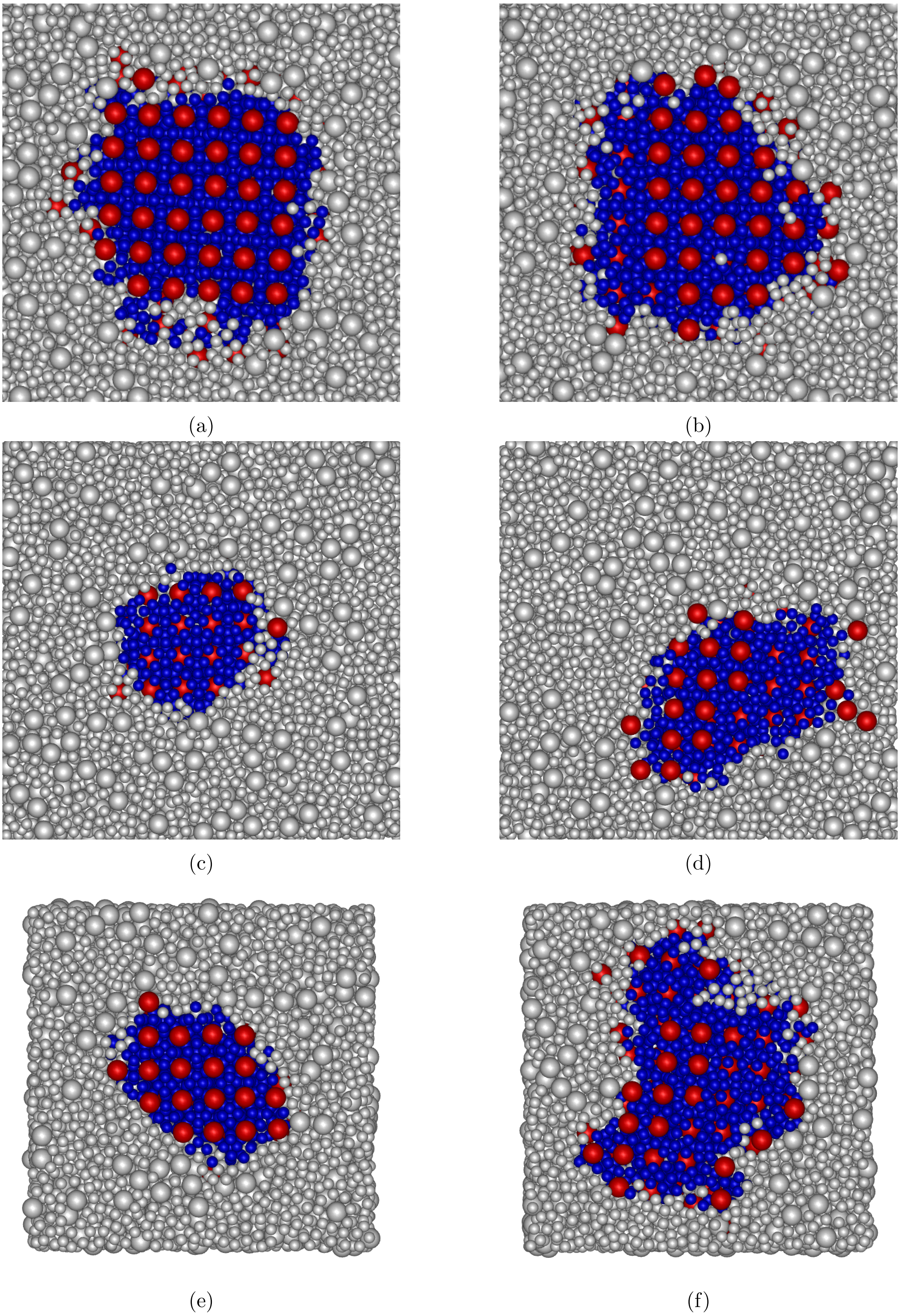
Figure 20: Number of particles with in the nucleus as a function of every 50 MC cycles at a constant temperature $k_B T/\epsilon = 0.025$ and constant pressure $\beta P \sigma_L^3 = 62.00$ for the L (red) and XL (blue) seeding configurations (a), and the S (blue) and L (red) seeding configurations in (b).

We observe from figure 20a that the size of the nucleus grows by roughly the same

amount for every 50th MC cycle for the large and extra large seeding configurations, this trend is also observed when comparing the small and medium seeding configurations as seen in figure 20b. This is an expected result as the size of the seeds used in the XL and L seeding configurations are the same size as well as in the case where the S and M seeding configurations are compared. We can not deduce the rate with which the AB_{13} crystals nucleate from the slope as the notion of time is not realized in a MC simulation.
Since the growth trajectories given in figure 20 certainly suggest that growth is taking place we then conducted a visual analysis as seen in figure 21. We take an initial snapshot and a final snapshot of the cross section of each system where the initial snapshot is the snapshot taken at the first MC cycle while the final snapshot is a snapshot of the nucleus at the final MC cycle as seen in figure 20. Due to the computational demands of the large and extra large seeding configurations it was not possible to generate more MC cycles. Consequently, there is not a noticeable difference in the size of the nucleus as seen in the initial snapshot (figure 21a) and the final snapshot (figure 21b) for the case of the extra large seeding configuration. This trend was also observed for the case of the large seeding configuration and is therefore not shown here.

However this is not the case for the small and medium seeding configurations. We display the initial and final snapshots of the medium seeding configuration in figure 21c and 21d respectively, we notice a distinct difference in the size of the nucleus. Similarly we find the same trend when we compare the initial and final snapshot in figure 21e and 21f respectively for the case of the small seeding configuration.

For the small seeding configuration it was found that only 14% of the system had nucleated while only 10% had nucleated in the case of the medium seeding configuration.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 21: Cross sections taken of the nucleus for the extra large seeding configuration, (a) the initial snapshot and (b) the final snapshot. Additionally for the nucleus of the medium sized seeding configuration in (c) the initial snapshot and (d) the final snapshot and finally the cross sections of the nucleus in the small seeding configuration (e) the initial snapshot and (f) the final snapshot.

7 Conclusions and outlook

It was found that for a temperature $k_BT/\epsilon = 0.025$ by means of thermodynamic integration the pressure of coexistence of the crystalline phase with that of a binary fluid phase was $\beta P_{coex}\sigma_L^3 = 45.38$. Although other temperature regimes were investigated namely $k_BT/\epsilon = 0.2$ and $k_BT/\epsilon = 0.1$, their respective pressure at which coexistence occurs was not determined but rather a pressure region was defined in which the coexistence pressure exists. For $k_BT/\epsilon = 0.1$ we found that the coexistence pressure existed in the region $\beta P\sigma_L^3 > 35.0$ and finally for $k_BT/\epsilon = 0.2$ we found the coexistence pressure in a region $\beta P\sigma_L^3 > 37.0$.

From the results presented in section 5.1 we found that we were able to classify the local structure of binary particles as either crystalline species A, crystalline species B (within the *ico*-AB$_{13}$ crystal structure), FCC or fluid phase using the $\bar{\omega}_8 - \bar{\omega}_6$ plane together with the diameter of the particle. Additionally it was found that in section 5.2 it was necessary to resort to the use of a neural network ensemble in order to obtain a network capable of producing satisfactory classifications. It was found that networks trained using the SGD optimizer performed better compared with networks trained using the Adam optimizer and this could be due to over fitting within the Adam networks. We could see this from the loss curves in figure 13 where the loss of the Adam method is approximately zero. Therefore neural networks trained using the Adam optimizer should be trained with less amount of epochs so as to avoid over fitting. Adam does however offer quicker training times as well as being the optimizer of choice when faced with limited data due to the rapid convergence of Adam networks to an accurate network.

Although the bond order parameters were sufficient for the identification of the local structure of a colloidal particle, the neural networks did offer additional perks. As seen in figure 17, essential order parameters necessary to make predictions of the local structure of colloidal particles are $\bar{\omega}_8$ and $\bar{\omega}_6$, which was corroborated by the visual analysis in figure 12. Unfortunately from the feature importance analysis one is unable to determine what feature is informative and what feature is redundant to the neural networks. Future networks may employ feature extraction whereby redundant features or features that posses little impact on the classification of the local structure of a particle may be removed as part of the input layer. It is expected that this would greatly improve the classifications of the neural network. This in turn could potentially allow us to reveal bond order parameters of significance within a system and in turn reveal the underlying symmetries. It is for this reason that neural networks are a preferable option in comparison with one manually analyzing the distributions of the values of the bond order parameters.

Since we were able to show nucleation of AB$_{13}$ crystals within a binary fluid of WCA particles, the results would suggest that a seed with radius r_S would indeed grow further for pressures $\beta P\sigma_L^3 > 62.00$, where r_S is the radius of the nucleus in the S configuration from table 4. The critical pressure is still yet to be determined for the critical radius size r_c. Once the critical pressure is calculated it would then be of interest to determine the nucleation barrier and the rate of nucleation of AB$_{13}$ crystals within binary mixtures.

Appendices

A Monte Carlo Simulation

Monte Carlo simulations (abbreviated as MC simulations) allow us to sample configurational phase space. In order to understand how MC simulations work we need to review statistical mechanics. We draw our attention to the case of an $\{NVT\}$ ensemble, where the number of particles N, the volume V and the temperature T are kept fixed. The partition function which describes this system is proportional to a Boltzmann factor $\exp(-\beta H)$ where H denotes the Hamiltonian of the system. The partition function is given by

$$Z(N,V,T) = \frac{1}{N!h^{3N}} \int d\mathbf{\Gamma}^N exp\left(-\beta H(\mathbf{\Gamma}^N)\right), \tag{16}$$

where $\mathbf{\Gamma}^N = \left(\mathbf{r}^N, \mathbf{p}^N\right)$ denotes the phase space points with position $\mathbf{r}$ and momenta $\mathbf{p}$ of the N particles, with h denoting the Planck constant and $\beta^{-1} = k_B T$ where k_B is the Boltzmann constant. An ensemble average of an arbitrary measurable equilibrium property denoted A is given by

$$\langle A \rangle = \frac{\int d\mathbf{\Gamma}^N A\left(\mathbf{\Gamma}^N\right) exp\left(-\beta H(\mathbf{\Gamma}^N)\right)}{\int d\mathbf{\Gamma}^N exp\left(-\beta H(\mathbf{\Gamma}^N)\right)}, \tag{17}$$

which reduces to

$$\langle A \rangle = \frac{\int d\mathbf{r}^N A\left(\mathbf{r}^N\right) exp\left(-\beta U(\mathbf{r}^N)\right)}{\int d\mathbf{r}^N exp\left(-\beta U(\mathbf{r}^N)\right)}, \tag{18}$$

in the case where property A is independent of momenta $\mathbf{p}$ and only dependent on the positions $\mathbf{r}$, here $U\left(\mathbf{r}^N\right)$ denotes the potential energy. Equation 18 implies that the average of quantity $\langle A \rangle$ can be computed as the weighted sum of quantity A for each configuration. Therefore we may express this average as

$$\langle A \rangle \approx \frac{1}{M} \sum_{i=1}^{M} A\left(\mathbf{r}_i^N\right), \tag{19}$$

where M denotes the total number of configurations that have been generated.

Let us now consider how we generate configurations in phase space. Firstly we generate an initial configuration $\mathbf{r}^N$ that has a non vanishing Boltzmann weight $p(\mu) = \exp(-\beta U_\mu)$, where this state is labeled μ. A new trial configuration (labeled μ') is then generated with a Boltzmann weight $p(\mu') = \exp\left(-\beta U_{\mu'}\right)$. This new configuration μ' is then rejected or accepted as a valid configuration based upon an "acceptance rule". This rule however must satisfy the detailed balance condition. The condition of detailed balance states that in equilibrium the average number of accepted trial moves from state μ to state μ' must be equal to the average number of accepted trial moves from state μ' to state μ. This implies the following

$$p(\mu)P(\mu \to \mu')acc(\mu \to \mu') = p(\mu')P(\mu' \to \mu)acc(\mu' \to \mu), \tag{20}$$

where $P(\mu \to \mu')$ denotes the probability of performing a trial move from state μ to state μ' and $acc(\mu \to \mu')$ denotes the probability of accepting the move from state μ to state μ'. As seen in the original Metropolis scheme the probability P is chosen to be symmetric, so $P(\mu \to \mu') = P(\mu' \to \mu)$. Using this fact together with equation 20 we find that the probability for a transition from state μ to state μ' to be accepted is given by

$$\frac{acc(\mu \to \mu')}{acc(\mu' \to \mu)} = \exp\left[-\beta\left(U_{\mu'} - U_{\mu}\right)\right]. \tag{21}$$

A random number is then generated from a uniform distribution in the range $[0, 1]$, should this number be less than $acc(\mu \to \mu')$ then the new configuration is accepted; however, if this random number is larger than $acc(\mu \to \mu')$ then the new configuration is rejected.

B Verlet List Algorithm

The distances between the particles needs to be calculated for every particle move in order to determine the move acceptances of future particle moves. This process of having to recalculate the particle distances for every particle move is computationally expensive as the computation time scales as N^2. Therefore we employ the use of the Verlet list algorithm which is outlined in many texts regarding MC simulations such as [2, 3]. We shall provide a short summary of how the Verlet list algorithm is implemented.

The Verlet list algorithm constructs a list of neighbors j for each particle i within the system. Here the neighbors are defined to be all particles within the distance $r_m = r_{cut} + r_{skin}$, where r_{cut} is the cutoff distance for the first shell of neighbors as characterized by the radial distribution function $g(r)$, and r_{skin} is taken to be $r_{skin} = 0.3\sigma$. An example of these cutoffs are given in figure 22 below.

When a particle move is accepted such that the displacement δx given by the particle move from its original position, is greater than r_{skin}, then the verlet neighborhood list is reconstructed[vii].

The Verlet list algorithm hinges on the idea that a large fraction of the neighbors of each particle remains the same. Therefore it seems unnecessary to recalculate all the distances between the particles for each MC particle move.

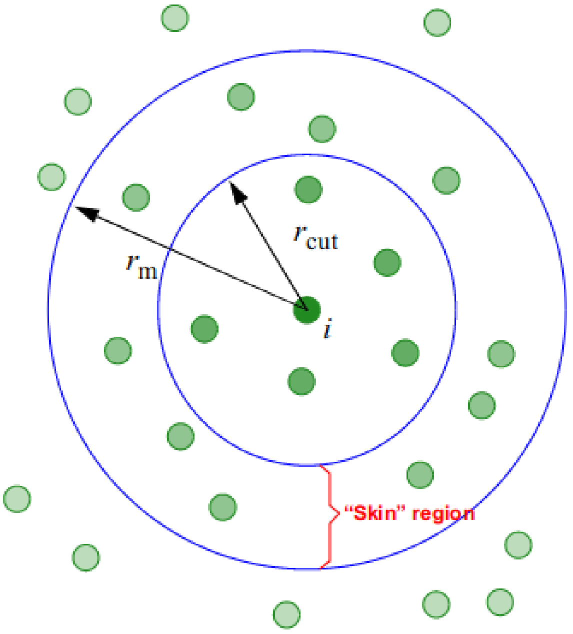


Figure 22: Figure showing an example of the cutoff radius r_{cut} and the verlet list cutoff radius r_m around a particle i . Source: `http://www.acclab.helsinki.fi/~knordlun/moldyn/lecture03.pdf`.

C SANN Algorithm

A parameter-free algorithm called the SANN algorithm is introduced in [25]. This algorithm allows us to identify the number of nearest neighbors for each particle i individually, based on a unique cutoff radius $R_i^{(m)}$. Unlike the fixed distance cutoffs determined from the radial distribution functions. The SANN

[vii] This condition applies for the case of MC simulations, in the case of molecular dynamic simulations the Verlet list is updated every 10-20 time steps.

algorithm method is simple; firstly, the algorithm ranks the particles j nearest to particle i sequentially such that $r_{ij} < r_{i,j+1}$ for all particles j. Now based on the distance between particle i and particle j we are then able to associate an angle θ_{ij}. The SANN algorithm sums the solid angles associated with the angle θ_{ij} until the sum equates to 4π. So

$$4\pi = \sum_{j=1}^{m} 2\pi(1 - \cos(\theta_{ij})). \tag{22}$$

Once this point is reached we are then able to define the number of nearest neighbors for particle i denoted here by m. Once the number of nearest neighbors for particle i has been determined the cutoff radius $R_i^{(m)}$ for particle i can then be calculated according to

$$R_i^{(m)} = \frac{\sum_{j=1}^{m} r_{ij}}{m-2}. \tag{23}$$

A drawback of the SANN algorithm is that it is not symmetric in the sense that particle i may be a nearest neighbor of particle j; however, particle j need not necessarily be a nearest neighbor of particle i. Therefore a simple way to remedy this issue is to disqualify a nearest neighbor if the two particles i and j are not mutually nearest neighbors.

References

[1] V. I. Kalikmanov. *Nucleation Theory.* Springer, Delft University of Technology, Delft, The Netherlands, 1 edition, 2013.

[2] W. Krauth. *Statistical Mechanics: Algorithms and Computations.* OXFORD University Press, Laboratoire de Physique Statistique, Ecole Normale Supérieure, Paris, 1 edition, 2006.

[3] D. W. Heerman K. Binder. *Monte Carlo Simulation in Statistical Physics.* Springer, Verlag Berlin Heidelberg, 5 edition, 2010.

[4] B. Smit D. Frenkel. *Understanding Simulation From Algorithms to Applications.* Academic Press, Harcourt Place, 32 Jamestown Road, London, United Kingdom, 2 edition, 2002.

[5] Rongjin Huang, Yanying Liu, Wei Fan, Jie Tan, Furen Xiao, Lihe Qian, and Laifeng Li. Giant negative thermal expansion in nazn13-type la (fe, si, co) 13 compounds. *Journal of the American Chemical Society*, 135(31):11469–11472, 2013.

[6] Seung Gu K and Kazuaki F. Magnetic properties of la (co1- xcrx) 13 compounds with an nazn13-type structure. *Journal of alloys and compounds*, 244(1-2):146–150, 1996.

[7] Elena V Shevchenko, Dmitri V Talapin, Stephen O'Brien, and Christopher B Murray. Polymorphism in ab13 nanoparticle superlattices: An example of semiconductor- metal metamaterials. *Journal of the American Chemical Society*, 127(24):8741–8747, 2005.

[8] AB Schofield, PN Pusey, and P Radcliffe. Stability of the binary colloidal crystals a b 2 and a b 13. *Physical Review E*, 72(3):031407, 2005.

[9] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance. *arXiv preprint arXiv:1801.01489*, 2018.

[10] Jorge R Espinosa, Carlos Vega, Chantal Valeriani, and Eduardo Sanz. Seeding approach to crystal nucleation. *The Journal of chemical physics*, 144(3):034501, 2016.

[11] Laura Filion, Michiel Hermes, Ran Ni, and Marjolein Dijkstra. Crystal nucleation of hard spheres using molecular dynamics, umbrella sampling, and forward flux sampling: A comparison of simulation techniques. *The Journal of chemical physics*, 133(24):244115, 2010.

[12] Urs Gasser, Florian Ziese, and Georg Maret. Characterization of local structures with bond-order parameters and graphs of the nearest neighbors, a comparison. *The European Physical Journal Special Topics*, 223(3):455–467, 2014.

[13] Pieter-Rein ten Wolde, Maria J. Ruiz-Montero, and Daan Frenkel. Simulation of homogeneous crystal nucleation close to coexistence. *Faraday Discuss.*, 104:93–110, 1996.

[14] Paul J Steinhardt, David R Nelson, and Marco Ronchetti. Bond-orientational order in liquids and glasses. *Physical Review B*, 28(2):784, 1983.

[15] Wolfgang Lechner and Christoph Dellago. Accurate determination of crystal structures based on averaged local bond order parameters. *The Journal of chemical physics*, 129:114707, 10 2008.

[16] Fernandes de Mello Rodrigo and Antoonelli Ponti Moacir. *Machine Learning: A Practical Approach on the Statistical Learning Theory.* Springer, Gewerbestrasse 11, 6330 Cham, Switzerland, 1 edition, 2018.

[17] Hastie Trevor, Tibshirani Robert, and Friedman Jerome. *The Elements of Statistical Learning.* Springer, Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA, 2 edition, 2009.

[18] Christopher M. Bishop. *Neural Networks for Pattern Recognition.* CLARENDON PRESS • OXFORD, Department of Computer Science and Applied Mathematics Aston University Birmingham, UK, 1 edition, 1995.

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[20] Navid Azizan and Babak Hassibi. Stochastic gradient/mirror descent: Minimax optimality and implicit regularization. *arXiv preprint arXiv:1806.00952*, 2018.

[21] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):993–1001, 1990.

[22] David W Opitz and Jude W Shavlik. Actively searching for an effective neural network ensemble. *Connection Science*, 8(3-4):337–354, 1996.

[23] M.D. Eldridge, P.A. Madden, P.N. Pusey, and P. Bartlett. Binary hard-sphere mixtures: a comparison between computer simulation and experiment. *Molecular Physics*, 84(2):395–420, 1995.

[24] Emanuele Boattini, Michel Ram, Frank Smallenburg, and Laura Filion. Neural-network-based order parameters for classification of binary hard-sphere crystal structures. *Molecular Physics*, 116(21-22):3066–3075, 2018.

[25] Jacobus A van Meel, Laura Filion, Chantal Valeriani, and Daan Frenkel. A parameter-free, solid-angle based, nearest-neighbor algorithm. *The Journal of chemical physics*, 136(23):234107, 2012.

[26] MJ Murray and JV Sanders. Close-packed structures of spheres of two different sizes ii. the packing densities of likely arrangements. *Philosophical Magazine A*, 42(6):721–740, 1980.

[27] P Bartlett, RH Ottewill, and PN Pusey. Superlattice formation in binary mixtures of hard-sphere colloids. *Physical Review Letters*, 68(25):3801, 1992.

[28] MD Eldridge, PA Madden, PN Pusey, and P Bartlett. Binary hard-sphere mixtures: a comparison between computer simulation and experiment. *Molecular Physics*, 84(2):395–420, 1995.

[29] X Cottin and PA Monson. Substitutionally ordered solid solutions of hard spheres. *The Journal of chemical physics*, 102(8):3354–3360, 1995.

[30] GA Mansoori, Norman F Carnahan, KE Starling, and TW Leland Jr. Equilibrium thermodynamic properties of the mixture of hard spheres. *The Journal of Chemical Physics*, 54(4):1523–1525, 1971.

[31] Karin Overgaag, Wiel Evers, Bart de Nijs, Rolf Koole, Johannes Meeldijk, and Daniel Vanmaekelbergh. Binary superlattices of pbse and cdse nanocrystals. *Journal of the American Chemical Society*, 130(25):7833–7835, 2008.