Monte-Carlo simulations of defect-rich tilings of polydisperse squares

Bachelorarbeit

eingereicht von

Felix Aron Dannert

im

Fachbereich Physik

an der

Universität Konstanz

Betreuer: J. Häring

Gutachter: Prof. Dr. M. Fuchs

August 2018

Abstract

Fachbereich Physik
Universität Konstanz

Bachelor's Thesis

Monte-Carlo simulations of defect-rich tilings of polydisperse squares

by Felix Aron Dannert

Two-dimensional systems have long been a subject of research, as they exhibit interesting behavior not found in three dimensions. The thesis presented here examines two-dimensional, hard core potential systems of squares. These squares are anisotropic and exclusively driven by entropic forces. Their equilibrium behavior is accessed through Monte-Carlo simulations. The different thermodynamic phases of this system are analyzed and the thesis reports an intermediate phase between the solid and the fluid called the tetratic phase [1]. Defects in the system are evaluated. It is found that the point defect density increases with the packing fraction to values of up to 1.4%. Finally, the dispersion relations of the lattice vibrations are calculated via an elasticity theory for real crystals [2]. Most interestingly the frequency of the angular vibrations in the solid phase does not tend to zero in the hydrodynamic limit. In the tetratic phase however, the same branch tends to zero in this limit case.

Deutsche Zusammenfassung

Fachbereich Physik
Universität Konstanz

Bachelorarbeit

Monte-Carlo simulations of defect-rich tilings of polydisperse squares

'Monte-Carlo Simulationen von defektreichen Kachelungen polydisperser Quadrate'

von Felix Aron Dannert

Zweidimensionale Systeme werden schon seit langem wegen ihrem interessanten, in drei Dimensionen nicht vorzufindendem Verhalten untersucht. Die hier präsentierte Arbeit widmet sich demnach zweidimensionalen Systemen von Quadraten, welche ausschließlich über ein, ihre Geometrie repräsentierendes, unendliches Kastenpotential wechselwirken. Das bedeutet, dass die Quadrate nicht überlappen können. Sie sind anisotrop und die Gleichgewichtseigenschaften des Systems werden mittels Monte-Carlo Simulationen betrachtet. Die unterschiedlichen auftretenden thermodynamischen Phasen werden analysiert. Es wird eine Phase zwischen der festen und der fluiden Phase gefunden: die sogenannte tetratische Phase [1]. Eine Studie der Defekte in der Kristallstruktur ergibt eine mit der Packungsdichte ständig steigende Dichte an Punktedefekten. Hierbei wird ein Maximalwert von 1.4% erreicht. Die Dispersionsrelationen der Gitterschwingungen werden über eine Elastizitätstheorie der realen Kristalle [2] berechnet. Es zeigt sich, dass in der festen Phase die Frequenz des Winkelzweiges der Dispersionsrelation im hydrodynamischen Limit nicht null ist. Interessanterweise geht diese Frequenz in der tetratischen Phase für den gleichen Grenzfall gegen null.

Contents

Chapter 1

Introduction

The two-dimensional system of anisotropic hard squares presents fascinating phase behavior [1, 3, 4] and dispersion relations while being easily accessible through simple Monte-Carlo simulations.

This thesis uses the Metropolis-scheme Monte-Carlo method to sample the bulk properties of squares confined to two-dimensional space. These squares have no interaction apart from their infinite hard core potential, implying that the squares can not overlap. Therefore the system is driven exclusively by entropic forces. The experimental counterpart to these simulations is the system of three-dimensional colloidal cubes sedimented into monolayers [5].

Two-dimensional systems are generally interesting for several reasons. Historically they are closely connected to the invention of Monte-Carlo simulations [6] and sparked a lengthy discussion about whether crystals can exist in less than three dimensions. One major contribution to this discussion was made by Mermin and Wagner by proving that, at finite temperatures, indeed no true crystal can exist in one or two dimensions [7]. This demeanor is maintained by long wave length, low energy modes, the so called Mermin-Wagner fluctuations. In contrast to the case in three dimensions, a crystal in two dimensions is not characterized by long range length scales in the positional order, but by only quasi-long range positional order [8].

The phases and phase transitions of two-dimensional systems are especially well understood. The phases are primarily distinguished by their internal order in respect to orientation of the neighbors of particles and position of the particles. For most systems in two dimensions there even exist microscopic theories of melting [9–11], which are yet to be found in three dimensions. The most popular one [12], and the one which this thesis will generally refer to, is called the KTHNY-theory. It incorporates the manifestation and behavior of crystallographic defects into the microscopic process of melting

[13].
The very same theory also predicts phases generally not present in three-dimensional systems. These x-atic phases are identified by having no order in the particle position, but only in the orientation of particle neighbors [1].
The characteristics discussed above could also be found in the much simpler system of hard disks [8, 14–16], so why is it worthwhile to consider hard squares?
The particular feature of the hard squares is their anisotropic hard core potential. It promises to influence the orientation of the particles and their surroundings and with that any behavior connected to these orientations. Furthermore, the system of hard squares crystallizes in a tetragonal lattice. Since this lattice type is quite uncommon, unusual methods have to be deployed in order to identify the local structure of the lattice.

The thesis aims to apply the elasticity theory for real crystals developed by M. Fuchs *et al.* [2] to the solid phase and the x-atic phase of the system of hard squares in order to compute dispersion relations of the lattice vibrations. Since this theory is developed to include crystallographic defects, an analysis of the latter shall be performed. It will also be necessary to perform a phase analysis so that the correct phases are analyzed for the dispersion relations.
One difficulty generally appearing in simulations is to gauge the influence of the finite size of the simulated systems on the results. A so called finite size analysis shall therefore be performed on the results obtained from the simulations.

The following, already present reports of research are closely connected to the system of hard squares. In 2004 Wojciechowski and Frenkel performed Monte-Carlo simulations on the system of hard squares, suggesting the existence of the x-atic phase [3]. Dijkstra *et al.* used molecular dynamics and Monte-Carlo simulations to examine three-dimensional hard cubes [17]. They observe a delocalization of point defects that will be reproduced in this thesis. In 2017 Glotzer *et al.* executed very large simulations on two-dimensional polygons, including hard squares [1]. The phases of the system found in this thesis shall be compared to their work.

Structurally, the thesis will start in Chapter 2 with presenting the theoretical background of the physics as well as the simulation techniques, introducing phase transitions, crystallographic defects and the basics of the elasticity theory for real crystals. In Chapter 3 the implementation and preliminary results of the Monte-Carlo simulation will be outlined, showing how the raw data used in the analysis is obtained.
As mentioned above, in order to allow the application of the elasticity theory for real crystals, the system is first analyzed in respect to the phase and defect behavior. Therefore, Chapter 4 will outline a phase analysis performed by visual observations, a numerical

method and pair correlation functions. Although all of these methods utilize the same physical properties, the comparison of the results will ensure the accuracy of the methods.

In Chapter 5, the different kinds of defects in the system will be presented. The point defect density of the system is then calculated through reciprocal space. Furthermore, different methods for the analysis of topological defects will be suggested.

Consequently, Chapter 6 contains the dispersion relation of the lattice vibrations calculated via the elasticity theory for real crystals.

Finally, a summary and an outlook into future work will be given in Chapter 7.

Chapter 2

Theoretical Background

In this section, the theoretical background needed to understand the behavior of the system is outlined.

2.1 Phases and Phase Transitions

Thermodynamical phases are a very intuitive concept. When continually varying some external parameter of a system, say for example the temperature, the system behaves in a similar manner for some time. But as the parameter crosses over a certain point, the behavior of the system changes drastically. The perhaps best known example for this is the melting of water. A measure of water at normal pressure and a temperature of $-5°C$ can be picked up by hand and does not deform visibly when pressing on it. If the same water is now heated to $+5°C$, it will not resist any pressure applied by hand and will deform almost instantly. The water changed phase in between $+5°C$ and $-5°C$. Interestingly there are some regions around these temperatures, where a further change of the temperature will not drastically alter the behavior of the water.[1] These regions are the different phases of water.

2.1.1 Definition of Phases

The explanation above is close to what is thermodynamically defined as a phase. The thermodynamical definition dictates that a phase is

> [...] a system or part of a system which is homogeneous and has definite boundaries. [18, p. 3]

[1] $5°C$ water and $20°C$ water handles almost the same.

What is meant by homogeneous is homogeneous with respect to macroscopic variables. In this thesis the following slightly different but in principle equivalent definition shall be used.
Microscopically, different phases distinguish themselves from each other by their symmetry. Because the simulations in this thesis will produce microscopic data[2], it will not be trivial to gain access to macroscopic variables. Better accessible however are indeed the symmetries of the system. The definition of phases as used in this thesis is diverted to the definition of phase transitions. Accordingly the definition of a phase is: A region in the space of direct observables in between phase transitions.

2.1.2 Phase Transitions

Since different phases are distinguished by their internal symmetry, some measure for this symmetry must be found. This is done by the consequent introduction of the order parameters.

Introductory Example The definition of order parameters is simple to write down. But simple definitions are often at risk of being glanced over without developing a good understanding of the problem. Since the concept of order parameters is fundamentally important in this thesis, the following example is given in an effort to introduce them thoroughly. Note that the example lies not very close to this thesis thematically, but is still very relevant in respect to the introduction of order parameters.
A system that is well known in statistical mechanics is the Ising model. The two-dimensional Ising model consists of spins ordered on a lattice in x and y-directions. These l spins σ_l are always parallel or antiparallel to the z-axis and they always have norm one (i.e. $\sigma_l = \pm 1$). They can interact through their respective magnetic moment. The solving of the Ising model shall not be discussed here, but rather the resulting phenomena. But be aware that this problem can be solved with approximations [19] as well as exactly [20][3].
Let $m = \langle \sigma_l \rangle$[4] describe the average magnetic moment per lattice position. By solving the Ising model with approximations, one can show that the magnetization m is dependent on the temperature T in the system. Above the so called Curie-temperature T_C it is

[2] This data mainly being the particle positions and orientations.
[3] The proofs for the following statements can also be found in [19].
[4] Here, $\langle\rangle$ denotes averaging over all lattice positions l.

always zero, and below this temperature it is non zero [19, p. 295]

$$m = \begin{cases} 0 & T_C < T \\ \pm\sqrt{3(T_C - T)/T} & T < T_C \end{cases}. \tag{2.1}$$

Macroscopically, below the Curie-temperature the system is ferromagnetic, showing a permanent non-zero magnetization. Above T_C it is paramagnetic. Remembering the thermodynamic definition of phases above, this means that the system has two different phases above and below T_C and a phase transition at T_C. Looking at m shows that $m = 0$ in the paramagnetic phase and $m \neq 0$ in the ferromagnetic phase. m constitutes what is generally known as an order parameter. This is indicated by the fact that $m = 0$ in one phase and $m \neq 0$ in another.

Definition of Order Parameters For this thesis it shall be sufficient to define an order parameter in the following way.
An order parameter is a variable, for example an average over particles, which is non-zero in one phase of the system, tends to zero as the systems approaches the phase transition and is then zero in the consequent phase.
This is exactly what m provides for the ferromagnetic and paramagnetic phases in the Ising model.
It is important to mention at this point that the given definition omits a very important aspect about order parameters, namely their connection to the symmetry in the system. It will be tried to outline this connection in Appendix C.

Additional Example An further important example for this thesis is an order parameter connected to reciprocal space. Consider an finite system of particles ordered on a two-dimensional lattice. If one takes the Fourier transform of the microscopic density (2.7), it will exhibit Bragg-peaks of finite amplitude. The Bragg-peak height in a system at the reciprocal lattice vector $\mathbf{g}$ is given by [21]

$$n_{\mathbf{g}} = \frac{1}{V} \left\langle \sum_{\mu=1}^{N} e^{-i\mathbf{g}\cdot\mathbf{r}_\mu} \right\rangle, \tag{2.2}$$

where $\mathbf{r}_\mu$ are the particle positions and $\langle\rangle$ denotes the average over many equilibrium configurations. Because of the periodic structure of the particle positions, $n_{\mathbf{g}} \neq 0$.
If through some means the system is now melted, is looses its periodic structure in the

particle positions. This also entails the loss of the Bragg-peaks. So here, $n_{\mathbf{g}} = 0$[5]. Therefore $n_{\mathbf{g}}$ is a classical order parameter of the solid phase.

Ehrenfest Classification Not all phase transitions show the same behavior. They can be categorized by the Ehrenfest classification. For thermodynamics it states the following. At all phase transitions, the thermodynamical potential is not continuous in all direct observables. The phase transition is called a phase transition of n-th order, if the n-th derivative of the thermodynamical potential with respect to one direct observable is not continuous and if all lower order derivations are. [22]
Second order phase transitions are also called continuous phase transitions.

Most systems have more than one order parameter, so the challenge in finding different phases in the system is often the challenge of identifying the correct order parameters. In the following section this will be done for the system of hard squares.

Local Order Parameters In the system at hand, a thorough analysis of the phase transitions needs to be done. For the calculation of the dispersion relation it is vital to know which phase the system is in, as the method of the calculation as well as the expected results change in the different phases. As will be discussed in the following Section 2.4, for the presented system, the so called KTHNY-theory suggests the possibility of the presence of a novel phase, called the *tetratic* phase. In order to correctly identify this phase, one has to acquire additional information from the order parameters. This information is related to the spatial range of the order. To access this information, the order of the system has to be defined locally by so called local order parameters.
Three distinct local order parameters are chosen to analyze the system [1]. This choice relates closely to the anisotropic geometry of the particles, as it includes evaluation of the orientation.
The first is the local **body orientation** order parameter. For the μth particle with orientation α_μ in respect to a unit axis (Figure 2.1), it is defined as [1]

$$\xi_k^\mu = e^{ik\alpha_\mu}. \tag{2.3}$$

This local order parameter depends on the choice of k, which should reflect the k-fold symmetry of the particles themselves.
ξ_k^μ always gives a point on the unit circle in the imaginary plane depending on α_μ and k. The influence of k is that α_μ and $\alpha_\mu + \frac{2\pi}{k}$ result in the same point. So this parameter simply measures the orientation of the single particles with respect to k-fold symmetry.

[5] For this definition of the Bragg-peaks, this is not true in every system, as $n_{\mathbf{g}}$ includes the height of the noisy background. But this background tends to zero for $N \to \infty$.

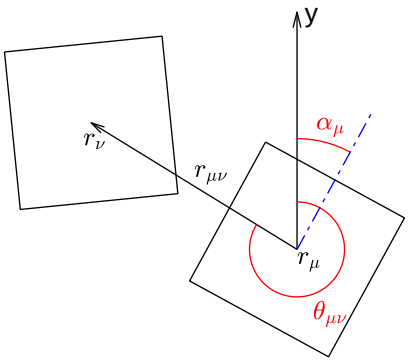

FIGURE 2.1: Depiction of the variables for the local order parameters. α_μ is the orientation of the μth particle, $r_{\mu\nu}$ is the bond vector between the centers of the μth and the νth particle and $\theta_{\mu\nu}$ is the bond angle between the μth and the νth particle.

This local definition can later be used in sophisticated analysis methods, but it also has a simple connection to the classical order parameters defined in the previous paragraphs. If the particles in the system have no preferred orientation, then the ξ_k^μ will be randomly distributed points on the unit circle in the imaginary plane. Averaging the value of ξ_k^μ over all $\{\mu\}$ will therefore yield zero. But if there exists a preferred orientation within the symmetry k, the distribution on the unit circle will not be random, but favor one direction. Here, the average will not be zero. Therefore $\langle \xi_k^\mu \rangle = \langle \xi_k \rangle$[6] is the order parameter of the body order for being only non-zero if the systems particle bodies are in an ordered phase. In such an ordered phase $\langle \xi_k \rangle$ is related to the direction of the average orientation of the particles.
In the system of hard squares, the particles have 4-fold symmetry. For ξ_k^μ in the following k is always chosen as $k = 4$, and $\xi^\mu := \xi_k^\mu$ is denoted.
The second local order parameter is the local **bond orientation** order parameter. It is defined as [1]

$$\psi^\mu_{k,\mathcal{N}_\mu} = \frac{1}{\mathcal{N}_\mu} \sum_{\nu \in \{\mathcal{N}_\mu\}} e^{ik\theta_{\mu\nu}}. \tag{2.4}$$

$\theta_{\mu\nu}$ is the angle between the vector connecting the particle μ to the particle ν and the y-axis (see Figure 2.1), $\mathcal{N}_\mu$ is the number of next neighbors of the μth particle and $\{\mathcal{N}_\mu\}$ is the index set of these neighbors. $\psi^\mu_{k,\mathcal{N}_\mu}$ works similarly to ξ_k^μ but does not regard the body orientation of a particle but rather the orientation of its neighbors. Since $\psi^\mu_{k,\mathcal{N}_\mu}$ already averages over its neighbors, with an analogue argumentation to the one made for ξ_k^μ the amplitude of $\psi^\mu_{k,\mathcal{N}_\mu}$ is small when the neighbors are oriented randomly and large if they obey k-fold symmetry. If then $\langle \psi^\mu_{k,\mathcal{N}_\mu} \rangle$ is taken, with the same argument it yields

[6] In this paragraph $\langle\rangle$ denotes the average over all particles in the system.

TABLE 2.1: The relation between the scaling of the spatial pair correlation function $C_O(r)$ and the order in the system. [14]

Scaling of C_O	Order
constant	long range
algebraic decay	quasi-long range
exponential decay	short range

zero if there is no order in the orientation of the configuration of next neighbors and is non-zero if there is. Therefore $\langle \psi^{\mu}_{k,\mathcal{N}_\mu} \rangle$ can be seen as the classical order parameter of bond orientation. Since the number of next neighbors and the symmetry of the system often coincides, define $\psi^{\mu}_{k} \coloneqq \psi^{\mu}_{k,\mathcal{N}_\mu}$ for $\mathcal{N}_\mu = k$. The hard square system will have four fold symmetry in its neighbors, so ψ_4 will be of major interest.
The third local order parameter is the local **positional** order parameter. It is defined as [1]

$$\chi^{\mu} = e^{i\mathbf{r}_\mu \cdot \mathbf{q}_0}. \tag{2.5}$$

Here, $\mathbf{r}_\mu$ are the particle positions and $\mathbf{q}_0$ is the first reciprocal lattice vector[7]. If the particle positions fit well on the periodic lattice given by the reciprocal lattice vector, the value of χ^{μ} will be close to one. Deviations from the periodic lattice rotate χ^{μ} on the unit circle in reciprocal space. So again if there is no positional order, the classical order parameter $\langle \chi^{\mu} \rangle$ is zero, if there is positional order, it is not.

2.1.3 Spatial Scaling of Local Order Parameters

Since the local order parameters are defined locally, it is possible to examine the spatial scaling of these parameters. This can, for example, be done with a spatial pair correlation function. This function is defined as the values $O^{\mu}\,(O^{\nu})^*$ plotted over $r_{\mu\nu} = |\mathbf{r}_\nu - \mathbf{r}_\mu|$ [1] with $O \in \{\xi, \psi_4, \chi\}$ [8]. It must be stressed here that these spatial correlation functions give access to the spatial behavior of the local order parameters. The following will heavily rely on the scaling of these spatial correlation functions. Through this scaling the range of the order in the system can be determined, as is shown in Table 2.1. Using this range of the order of the different local order parameters in the system, different internal symmetries and hence different phases can be identified. In the system of hard squares, three different phases are expected [1], the solid, the tetratic and the liquid phase. Their definitions by the order are presented in Table 2.2.

[7] The first reciprocal lattice vector can be seen as the position of the highest Bragg-peak (not the constant peak) in the Fourier transform of the microscopic density.
[8] Formal definitions for the spatial correlation functions are given in Section 4.5.

TABLE 2.2: The range of the order defines different phases. The phases listed are those to be expected in the system of hard squares. [8]

Phase	Order in χ	Order in ψ_4
solid	quasi-long range	long range
tetratic	short range	quasi-long range
fluid	short range	short range

There are some interesting aspects about these definitions. First, from three-dimensional systems one is used to the positional order being long range in a solid phase. But in Table 2.2, the χ order for the solid phase is given as quasi-long range. This stems from a specialty in two dimensions explained in Section 2.2, the Mermin-Wagner fluctuations. These fluctuations destroy any long range positional order in the system.
But why then is it possible for the orientational order ψ_4 to have a long range order? This is because the argument of Mermin and Wagner is only applicable for continuous symmetries, which the orientational symmetry is not. The reasoning for the exclusive applicability on continuous symmetries is also laid out in Section 2.2.
Also special here is the tetratic phase, because it only appears in two-dimensional systems. It can be characterized as a phase where the positional crystalline order has been lost, but a reduced order in the orientation is still present. These scalings of the local order parameters will be the basis for the phase analysis presented later in this thesis.

2.2 Mermin-Wagner Fluctuations

As has already been mentioned, a specialty of two- and one- dimensional systems are the Mermin-Wagner fluctuations [23]. They are low energy, long wave lengths modes that can be excited at any temperature $T > 0$. In the special case of two-dimensional particle systems, the argument from Mermin and Wagner is derived from the Bogoliubov inequality and uses transformations of the continuous positional symmetry in the particle positions. The argument leads to the final result [7]

$$n_{\mathbf{k}} \lesssim \frac{1}{(\ln(N))^{\frac{1}{2}}}. \tag{2.6}$$

$n_{\mathbf{k}}$ is defined like $n_{\mathbf{g}}$ in equation (2.2) only for all reciprocal wave vectors $\mathbf{k}$ and N is the number of particles in the system.
The inequality (2.6) is particularly true for the height of the Bragg-peaks $n_{\mathbf{g}}$, so it states

that for infinitely large two-dimensional particle systems, all Bragg-peaks vanish[9]. But since Section 2.1.2 identifies the height of the Bragg-peaks $n_{\mathbf{g}}$ as the order parameter of the solid phase, this implies that these systems can not transition into a truly solid phase.
As can be seen in the previous section, structures that are locally very similar to solids can still be identified in two dimensions. Their spatial range in positional order is quasi-long range.

2.3 Crystallographic Defects

Although they are often low in concentration, defects in the structure of crystalline lattices can be responsible for a good portion of the behavior of the crystal. In the two-dimensional system of hard squares this is especially true, as Section 2.4 will connect defects to the transition of phases. Ashcroft and Mermin define a crystallographic defect as

> [...] any region where the microscopic arrangement of ions [here: particles] differs drastically from that of a perfect lattice. [24, p. 616]

These deviations from the perfect lattice can be classified into many different categories. The two most relevant for this thesis are presented in the following.

2.3.1 Point Defects

Point defects constitute a class of crystallographic defects that consist exclusively of interstitials and vacancies.
A vacancy is defined as a missing particle at a lattice positions. An interstitial is defined as an additional particle in between lattice position. See Figure 2.2 for visual reference.
Point defects can create distortions in the lattice around them, but the displacement fields are of finite range. Therefore they do not affect the global order of the system.

2.3.2 Topological Defects

Another class of defects that is important for the system of hard squares is the class of topological defect. Formally topological defects are defined as defects where single

[9] In this thesis, the height of the Bragg-peaks will be used regularly. They only appear because of the finite size of the system. This is an indicator for the fact that some of the analysis applied relies on the finite size of the system or at least on the finite size of the sample.

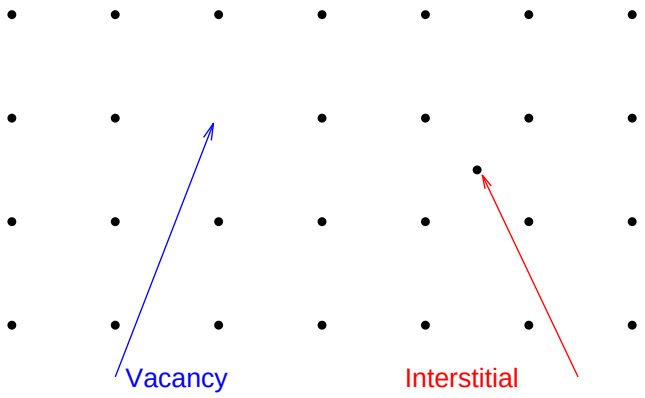

FIGURE 2.2: Examples for a vacancy and an interstitial in a *sc*-lattice. The blue arrow indicating the vacancy shows a lattice position with a missing particle. The red arrow indicating the interstitial shows an additional particle between lattice positions.

occurrences can not be removed from the system by continuous transformations [13]. This definition is not very easy to understand, but Chaikin and Lubensky offer the following explanation. This explanation takes the example of a dislocation, a type of topological defect (see Figure 2.3) and states the following. Since at the dislocation an inserted lattice line originates, in order to remove the topological defect one would have to remove the inserted lattice line. But to do so particles at an arbitrary long distance away would have to be affected. So there can not be a transformation that continuously changes the periodic function describing the lattice while removing the inserted lattice line. [21]

Burgers-Vector Construction Later in this thesis, it will be tried to microscopically identify topological defects in simulations. The definition given above is not very useful for solving this problem. If there is a topological defect present in a lattice, typically there are also inserted lattice lines present. These should be detectable in the local lattice structure. The so called Burgers-vector construction uses this and offers a different, computationally more approachable definition. For the example of the dislocation shown in Figure 2.3, the following will demonstrate how different regions of the system are assigned a Burgers-vector. First, one has to choose an integration path. This integration path must go along the bonds between the particles in the local lattice structure and must be closed in a lattice without topological defects. The underlying *sc*-structure in Figure 2.3 usually leads to the choice of an integration box with same lengths in opposing sides. Now the starting position of the integration path is noted and the integration path is followed along on the local lattice structure. Then the end point of the integration path is noted and the vector connecting the two points is calculated. This vector is the

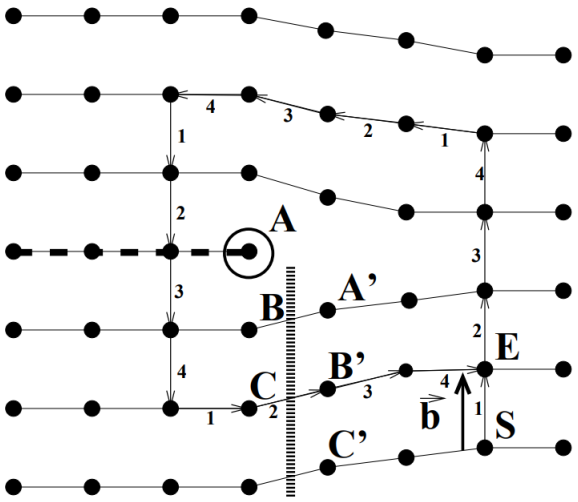

FIGURE 2.3: A dislocation in a *sc*-lattice as an example for a topological defect. The dislocation is the origin of an inserted lattice line starting at **A**. Also shown is the construction of the Burgers-vector. Starting from point **S**, four steps are taken in each unit direction of the lattice along the local lattice structure. This integration path ends at point **E**. The Burgers-vector is spanned between the points **S** and **E**. Figure extracted from [13].

Burgers-vector **b** (see Figure 2.3). If there is no inserted lattice line in the integration path, the start and the end point will coincide, yielding $\mathbf{b} = \mathbf{0}$. However if there is a inserted lattice line within the integration path, the start and end points will not coincide and the Burger-vector $\mathbf{b} \neq \mathbf{0}$ will be non zero[10]. In Section 5.2.1 it will be discussed that inserted lattice lines can only start and end in topological defects, so it can be assumed that inside an integration path yielding a non-zero Burgers-vector there must lie a topological defect. This will be the definition used for topological defects for the analysis of the same in real space.

One has to handle this redefinition of the topological defect with great care, as it comes at a price. An effect that is usually defined over the global behavior of the system is reduced to a microscopic definition. Surely, the definition of the Burgers-vector contains some amount of information on the behavior in the vicinity of the defect, but it can, for example, not describe how long an inserted lattice line travels before being eliminated by another topological defect or by a surface. This makes the definition less discrete and in some way it even loses the original meaning of the topological defect: the impact on the whole system. Therefore, even though it is far more complicated, the analysis of topological defect in the reciprocal space is superior to the microscopic analysis, because it automatically integrates the whole system into the discussion. Therefore for topological defects found this way, there is little room for arguing about the validity of

[10]The behavior is different if there are more than one inserted lattice lines. This is explained in the next paragraph.

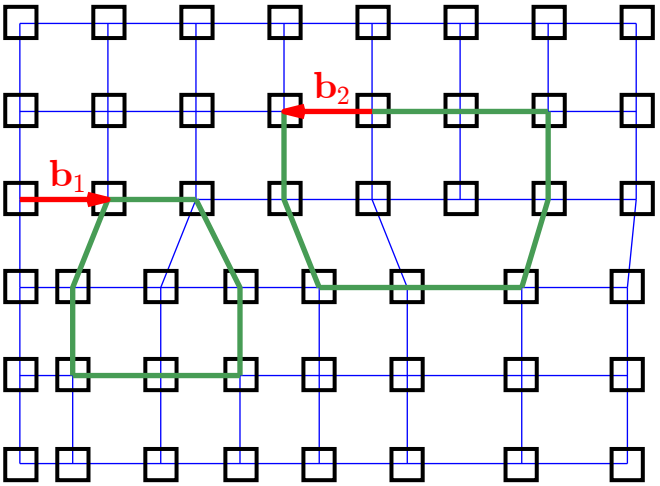

FIGURE 2.4: Dislocations with opposite Burgers-vectors. Depicted is a *sc*-lattice with two dislocations. The dislocations have opposite Burgers-vectors (in red), hence $\mathbf{b}_1 = -\mathbf{b}_2$. The construction of the Burgers-vectors via the integration path (in green) is explained in section 2.3.2. Both topological defects can be annihilated by reconnecting the bonds between the top and the bottom half of the lattice accordingly.

the defect. This is why the redefinition performed here must not be seen as absolute, but rather as a means to an end.

Formation of Topological Defects The previous paragraph states that topological defects can not be eliminated by continuous transformations. The same is true for the formation of topological defects. But how can a system then form or remove topological defects? The answer here is that while it is not possible to form or remove single defects, it is very well possible to form or remove pairs of defects. The Burgers-vector has an additional functionality apart from detecting defects. If two defects have opposing Burgers-vectors, so $\mathbf{b}_1 = -\mathbf{b}_2$, they can annihilate and remove the topological defect from the lattice (see Figure 2.4). In the same way, a fluctuating lattice can create two topological defects with opposing Burgers-vectors.

2.4 KTHNY-Theory

For some systems in two dimensions there exists a microscopic melting theory, the KTHNY-theory[11]. Since it will often be referenced in this thesis, it shall be briefly outlined[12]. The KTHNY-theory connects the appearance and behavior of topological defects to the phase transitions that the system performs.
As discussed in the previous section, dislocations first appear in bound dislocation pairs. This happens when the system is still in the solid phase. As the system now approaches its melting point these dislocation pairs begin to dissociate, changing the properties of the system. This leads to the loss of positional order in the system and therefore to a phase transition. This phase transition is continuous. Interestingly, the orientational order of the system is not lost in this process, and therefore there exists a phase between the solid and the liquid phase, generally called the x-atic phase[13]. Similarly, if this phase approaches its melting point, the dislocations dissociate into disclinations, another type of topological defects, and this causes the loss of orientational order. The system then continuously melts into the fluid phase. [13]

2.5 Elasticity Theory for Real Crystals

All real crystals exhibit point defects. Yet there used to be no microscopic elasticity theory taking these defects into account. This incorporation of defects in the periodic structure of the crystal is done by C. Walz and M. Fuchs in their elasticity theory for real crystals [2]. The following will briefly outline the argumentation given in [2] to arrive at the dispersion relation of the lattice vibrations through a reciprocal wave equation.

2.5.1 Basic Relations

In order to obtain the wave equation, first some basic relations must be introduced. Consider a system of N particles with mass m. The particle density is given by $n_0 = \frac{N}{V}$. The microscopic density of the crystal in real space is given by

$$\rho(\mathbf{r}, t) = \sum_{i=1}^{N} \delta(\mathbf{r} - \mathbf{r}_i(t)), \tag{2.7}$$

[11] After Kosterlitz, Thouless, Halperin, Nelson and Young.
[12] A more detailed summary can be found in [13].
[13] In the case of the hard squares, the underlying symmetry is tetragonal. Accordingly this intermediate phase is here called the tetratic.

where N is the number of particles in the system and $\mathbf{r}_i(t)$ are the time dependent particle positions. Applying the Fourier transformation to the density (2.7) yields the reciprocal density

$$\rho(\mathbf{k},t) = \int \mathrm{d}^d r\, e^{-i\mathbf{k}\cdot\mathbf{r}}\rho(\mathbf{r},t) = \sum_{i=1}^{N} e^{-i\mathbf{k}\cdot\mathbf{r}_i(t)}, \tag{2.8}$$

using properties of the Dirac delta function δ and the reciprocal wave vector $\mathbf{k}$. Similarly, via the spatial components α of the momenta p_i^α[14] the momentum density is defined as

$$j^\alpha(\mathbf{r},t) = \sum_{i=1}^{N} p_i^\alpha \delta(\mathbf{r}-\mathbf{r}_i(t)).$$

Again, the Fourier transformation yields the reciprocal momentum density

$$j^\alpha(\mathbf{k},t) = \sum_{i=1}^{N} p_i^\alpha e^{-i\mathbf{k}\cdot\mathbf{r}_i(t)}.$$

2.5.2 Zwanzig-Mori Equations of Motion

The Zwanzig-Mori formalism uses projection operations and identification of so called 'slow variables' to obtain equations of motion via the generalized Langevin equation. Note that the reciprocal lattice vectors are given by $\mathbf{g}$ such that $\mathbf{k} = \mathbf{g} + \mathbf{q}$, with $\mathbf{q}$ the reciprocal wave vector in the first Brillouin-zone. If the slow variables are chosen as the conserved momentum density $\delta j^\alpha(\mathbf{q},t)$ and the density fluctuations close to Bragg-peak positions $\delta n_{\mathbf{g}}(\mathbf{q},t)$[15] the Zwanzig-Mori formalism produces the equations of motion for an isothermal, dissipationless crystal. They are [2]

$$\frac{\partial}{\partial t}\delta n_{\mathbf{g}}(\mathbf{q},t) = \frac{-i}{mn_0} n_{\mathbf{g}}(g+q)^\alpha \delta j^\alpha(\mathbf{q},t) \qquad \text{and} \tag{2.9}$$

$$\frac{\partial}{\partial t}\delta j^\alpha(\mathbf{q},t) = -i\sum_{\mathbf{g}'\mathbf{g}} (g'+q)^\alpha n^*_{\mathbf{g}'} \mathbf{J}^*_{\mathbf{g}'\mathbf{g}}(\mathbf{q})\delta n_{\mathbf{g}}(\mathbf{q},t). \tag{2.10}$$

The * indicates the complex conjugate, $n_{\mathbf{g}}$ are the Bragg-peaks height order parameters defined in Section 2.1.2 and $\mathbf{J}^*_{\mathbf{g}'\mathbf{g}}(\mathbf{q})$ is the scaled inverse reciprocal density fluctuation correlation matrix defined as

$$\mathbf{J}_{\mathbf{g}'\mathbf{g}}(\mathbf{q}) = Vk_BT\left\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\right\rangle^{-1}.$$

k_B is the Boltzmann-factor and T is the temperature of the system.

[14] In the following superscript Greek letter mark spatial components.
[15] The reasoning behind this choice is presented in [2].

2.5.3 Wave Equation

A simple derivation now uses the equations of motions above to yield a wave equation. Differentiating equation (2.10) with respect to time, one receives

$$\begin{aligned}\frac{\partial^2}{\partial t^2}\delta j^\alpha(\mathbf{q},t) &= -i\sum_{\mathbf{g'g}}(g'+q)^\alpha n^*_{\mathbf{g'}}\mathbf{J}^*_{\mathbf{g'g}}(\mathbf{q})\frac{\partial}{\partial t}\delta n_{\mathbf{g}}(\mathbf{q},t)\\ &\overset{(2.9)}{=} -\frac{1}{mn_0}\sum_{\mathbf{g'g}}(g'+q)^\alpha n^*_{\mathbf{g'}}\mathbf{J}^*_{\mathbf{g'g}}(\mathbf{q})n_{\mathbf{g}}(g+q)^\beta\delta j^\beta(\mathbf{q},t)\\ &= -\frac{1}{mn_0}\Lambda^{\alpha\beta}(\mathbf{q})\delta j^\beta(\mathbf{q},t). \qquad (2.11)\end{aligned}$$

This is a wave equation in reciprocal space. In the hydrodynamical limit $\mathbf{q}\to 0$ it should give a classical wave equation in real space, so that the classical elasticity theory is a limit case of the presented theory. Using the property of the Fourier transformation stating that the transformation of a derivation is equal to a multiplication with the wave vector, one can show that for $\mathbf{q}\to 0$, $\Lambda(\mathbf{q})$ needs to have quadratic scaling.
Because of equation (2.11), the eigenvalues of

$$\Lambda^{\alpha\beta}(\mathbf{q}) = \sum_{\mathbf{g'g}}(g'+q)^\alpha n^*_{\mathbf{g'}}\mathbf{J}^*_{\mathbf{g'g}}(\mathbf{q})n_{\mathbf{g}}(g+q)^\beta$$

give the frequency $\omega^2(\mathbf{q})$ and therefore the dispersion relation.

2.6 Monte-Carlo Simulations

The Monte-Carlo method fundamentally constitutes a method for the solving of problems, or more precisely integrals, by applying statistical sampling. In this thesis however, the Monte-Carlo method is applied to a physical system to obtain information about that system in an equilibrium state. This is then called a Monte-Carlo simulation[16]. Here one randomly samples points from the equilibrated systems phase space to compute the stationary probability density of this system. One of the advantages of the Monte-Carlo simulation, at least for systems as simple as the one presented here, is the straightforward theory supporting this method. Only a few basic concepts have to be understood in order to produce the core of the Monte-Carlo algorithm.

[16] The invention of the Monte-Carlo method took place in the Manhattan-project and was therefore kept a secret. The code name Monte-Carlo was assigned, referring to the Monte-Carlo casino in Monte-Carlo, Monaco and with that joking about the connection of the method to chance. [25]

2.6.1 Detailed Balance

The first of these concepts is the condition of detailed balance. Suppose one considers a phase space containing the points a, b and c with the special attribute that one can only transition to points b and c starting form point a. Let $p(a)$, $p(b)$ and $p(c)$ be the stationary probability distribution of these points. Also denote $\mathcal{P}(x \to y)$ as the probability of moving from point x to point y. Since a move starting from a can only end on a itself or on b or c it must hold true that[17]

$$\mathcal{P}(a \to b) + \mathcal{P}(a \to c) + \mathcal{P}(a \to a) = 1. \tag{2.12}$$

But also one can quickly realize that because moves resulting in a must originate in a, b or c,

$$p(a) = p(a)\mathcal{P}(a \to a) + p(b)\mathcal{P}(b \to a) + p(c)\mathcal{P}(c \to a).$$

Combining this with (2.12) yields

$$\begin{aligned} p(a)\mathcal{P}(a \to b) + p(a)\mathcal{P}(a \to c) &= p(a)\left(1 - \mathcal{P}(a \to a)\right) \\ &= p(b)\mathcal{P}(b \to a) + p(c)\mathcal{P}(c \to a). \end{aligned}$$

This condition can be extended to real systems and then states for a point in phase space x and all the points reachable from that point in a move y_i

$$p(x)\sum_i \mathcal{P}(x \to y_i) = \sum_i p(y_i)\mathcal{P}(y_i \to x). \tag{2.13}$$

Now this expression can be easily written down, but is not as easily checked for simulations. A sufficient condition for (2.13) is the much easier checked *detailed balance* condition

$$p(x)\mathcal{P}(x \to y_i) = p(x)\mathcal{P}(y_i \to x) \quad (\forall\, i).$$

It should be stressed at this point that this condition is much stronger than (2.13) and does not have to be fulfilled in order to acquire a correct Monte-Carlo simulation. [6]

[17] The transition probabilities can be normalized to one.

2.6.2 Metropolis Algorithm

Now one has to think about how to fulfill detailed balance in a Monte-Carlo simulation. One possibility is to use the Metropolis algorithm. This algorithm simply give a transition probability that ensures that detailed balance is always satisfied. The Metropolis transition probability is [26]

$$\mathcal{P}(x \to y) = \min\left(1, \frac{p(y)}{p(x)}\right). \tag{2.14}$$

The two possible cases $p(x) < p(y)$ and $p(x) \geq p(y)$ quickly show that this transition probability always satisfies detailed balance.

2.6.3 Sampling

In the preceding section, often a move between points in the phase space has been referred to. Now the question of what such a move consists of will be answered.
In Monte-Carlo simulations, random points in the phase space are sampled. This sampling can be done in different ways, two of which will be presented here.

Direct Sampling The most straight forward route is to sample the phase space directly. Here points of the phase space are chosen randomly and independently from each other. This has the advantage that the snapshots[18] acquired this way are not correlated and can all be used to calculate the stationary probability density. But in application to the system at hand, this method has a big drawback. For the system of squares the phase space consist of all configurations, where none of the squares overlap. The phase space can be directly sampled by choosing all positions and rotations of the squares randomly. For low packing fractions, this random placement mostly results in points that belong to the phase space. But as the packing fraction increases, the direct sampling results in more collisions. If a configuration containing overlaps is generated this configuration is not a point in the phase space, hence $p(y) = 0$. Proceeding according to the Metropolis algorithm (2.14), the move resulting in a point of probability 0 must have a transition probability of 0. Therefore this sampling move is not accepted. For high packing fractions only very few direct sampling moves result in an actual phase space point, making the technique computationally expensive. [6]

Markov-Chain Sampling To achieve higher speeds for high packing fractions, the phase space can be sampled by Markov-Chain sampling. Here, starting from a point in

[18] Stills of the system configuration in equilibrium.

the phase space, a new sampling point is created by moving in a random direction by a random amount. In the system at hand such a move will consist of randomly rotating and displacing one square by an amount chosen from an uniform distribution between zero and a maximum amount. This can be a more efficient way of choosing samples. For the system of squares one can see that by starting from a configuration with no overlaps and only displacing and rotating one square by a small amount, one is quite likely to not produce a collision.
The drawback of this method is that the configurations produced by this way of sampling are most certainly correlated. Correlated configurations have the special property that they have a 'memory' of each other, meaning that if one of two correlated configurations is sampled, there is a probability higher than that corresponding to the stationary probability density that the second configuration will also be sampled. This means that these two correlated configurations contain less information about the stationary probability density than two decorrelated snapshots, because they cover less volume in the phase space.

2.6.4 Ergodicity

Because of the concept of correlated configurations it is important to look at the concept of ergodicity. An algorithm sampling a system is said to be ergodic if it can sample all points of the phase space in finite time. It is possible to construct algorithms that do not behave this way. These algorithms often do not produce sufficiently decorrelated configurations, making it hard to acquire the stationary probability density through them.

2.6.5 Ensemble Averages

Ensemble averages of some variable S are often the subject of discussion in statistical mechanics. They are given by

$$\langle S \rangle = \frac{\int S(\Gamma)\rho(\Gamma)\mathrm{d}\Gamma}{\int \rho(\Gamma)\mathrm{d}\Gamma},$$

with ρ the stationary probability density and Γ the points in the phase space. Here, the phase space is continuous. If the Monte-Carlo sampling yields N discrete points in the phase space $\{\tilde{\Gamma}_i\}$ that are dense in Γ, then the average can be approximated by

$$\langle S \rangle \approx \frac{1}{N}\sum_{i=1}^{N} S(\tilde{\Gamma}_i). \tag{2.15}$$

This also shows the importance of ergodicity, as approximation (2.15) can be very false for non ergodic sampled phase spaces. [27]

2.6.6 Periodic Boundary Conditions

In condensed matter physics, one is often interested in the bulk behavior of a system or material. When this behavior is to be simulated on a computer, the question arises of how to best model the system in a way that will represent the bulk.
The often times easiest implementation of boundaries is a box consisting of fixed walls. At points where the system is close to said walls it shows a disturbance of the bulk behavior by the surface. This disturbance might manifest in higher defect concentrations or even a completely different order. [6]
For very large systems this is not a problem, since enough volume of the system is far removed from the surface and does therefore show bulk behavior. In simulations however, the size of the system is always limited by the simulation time available. It is therefore often the case that the system is small enough for surface effects to be present throughout most of the system. Even worse is the fact that the ratio between system volume and surface always scales like the reciprocal diameter. This means that very small systems are especially at risk of showing behavior very much dictated by surface effects.
To circumnavigate this problem even in small systems, one can introduce periodic boundaries to the system. A periodic boundary is implemented by connecting the opposing sides of the simulation box. In this way, a two-dimensional square simulation box evolves to the surface of a torus. Since this can be quite abstract to visualize, one can also think of the system as an infinite repetition of the same simulation box. This solves the problem of unwanted surface effect because no preexisting surface is placed in the system. But this solution also produces some new problems. Close to phase transitions the correlation length of some variables in the system diverge. Implications for the periodic boundary system in this state are interesting because the correlation length can become bigger than the size of the system, meaning that particles can correlate with their periodic image.
In some applications one also has to be mindful of situations like the following. A particle a is moving strictly away from a resting particle b and neither its direction nor its velocity are ever changed. Without periodic boundaries, particle a can not collide with particle b, with periodic boundaries, it can.
Another very important effect of the finite size of the system coupled with the periodic boundary conditions is the restriction of reciprocal wave vectors. By solving the equation of motion it can be easily retraced that N particles can only have N discrete

modes in the first harmonic. So there are only N reciprocal wave vectors in the first Brillouin-zone. [22]

2.6.7 Finite Size Effects

Many averages and behaviors of a system can depend on its size. This is already discussed in Section 2.2 for the existence of Bragg-peaks in finite size two-dimensional systems. Therefore it is always important to look at the behavior of any results in respect to the system size. In this thesis this is done in the following way. Each result is calculated in systems of different sizes. For example if the point of a phase transition is acquired, it is done so for four different system sizes N. Then one of two cases can be seen. In the better case, the values converge onto a certain constant value for increasing system sizes. Then it can be inferred that this constant value is the value in the thermodynamic limit $N \to \infty$. Since the systems in this thesis are so small, this is unfortunately never the case. The analysis in this thesis is marked by the worse case, where no convergence for high N can be seen. In this case it must suffice to state that there are finite size effects present and the respective findings are not necessarily true in the thermodynamic limit $N \to \infty$.

Chapter 3

Implementation of the Monte-Carlo Algorithm

The Monte-Carlo algorithm will be implemented in accord to Section 2.6. A pseudocode of the Monte-Carlo algorithm can be found in the appendix E.

3.1 System Definition

The system consists of N squares with radii R_i which can move in a square periodic box of length L in two-dimensional space. With translation and rotation, a single square has three degrees of freedom. The definition of the single squares will be in accord with Figure 3.1. The inherent D_4 symmetry of the squares dictates that the states α_i and $\alpha_i + \frac{\pi}{2}$ are equivalent.

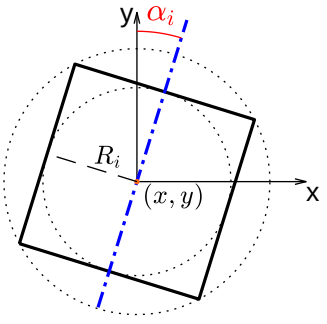

FIGURE 3.1: Geometric definition of a square. The position of the square is denoted by its center (x, y). The rotation α_i describes the angle between the normal of the uppermost face and the y-axis. The square inscribes a circle of radius R_i and is itself inscribed in a circle of radius $\sqrt{2}R_i$.

The radii R_i of the squares are chosen both monodispersely, meaning R_i is equal for all particles, and polydispersely, meaning R_i differs for each particle and is chosen uniformly in an interval $[R_{\min}, R_{\max}]$. The polydisperse case is especially interesting for experimentalists, as future experiments with three-dimensional neighborite cubes sedimented into monolayers might also possess polydispersity in the particle radii [5].

3.2 General Description and Detailed Balance

Because the system only has a hard core potential, all stationary probability densities $p(x_i)$ have the same value [6].[1] Using the Metropolis-scheme ensures detailed balance and also yields a very simple transition probability matrix

$$\mathcal{P}(x_i \to y_i) = \begin{cases} 1 & p(y_i) \neq 0 \\ 0 & p(y_i) = 0 \end{cases}.$$

This simply means that a move is accepted if there are no overlaps and rejected if there are.
For the reasons described in Section 2.6.3, a Markov-chain algorithm is used for sampling. As can be seen in the algorithm in appendix E, the course of action is the following.

1. Select particle with index z at random.
2. Decide randomly whether an oriantational (a) or translational (b) move will be performed.
   (a) Rotate the zth particle with orientational trial move algorithm given in Section E.1. The maximal rotation is controlled by Δ_α.
   (b) Displace the zth particle by a random amount in x-direction chosen uniformly from $[-\Delta_r, \Delta_r]$ and a random amount in y-direction chosen uniformly from $[-\Delta_r, \Delta_r]$.
3. If the zth square does not overlap with any other squares, continue to 1. If it does, revert the trial move and then continue to 1.

This algorithm is suggested in [6]. The only aberration is the probabilistic choice of what sort of move to perform. This was suggested by [27]. It gives the advantage that acceptance rates of the trial moves can be calculated for both move types separately.

[1] Of course the phase space of the system is continuous, but sampling discrete points in the Monte-Carlo simulation still allows the assignment of discrete non-zero probabilities $p(x_i)$. They can be interpreted as probability densities integrated over the volume surrounding x_i.

Since the acceptance rate for trial moves is optimal when it is about 20% [27, p. 46], this can now be optimized for both types of moves.

3.3 Collision Detection

A major effort in the Monte-Carlo routine will be to detect if the displaced or rotated squares will overlap with any other squares after the Metropolis move.
Luckily, the squares are convex bodies, and the employment of the Hahn-Banach theorem will solve the problem of collision detection by yielding the separating axis theorem (SAT). The SAT states the following:

> Two convex polygons are not colliding if there exists a line (or axis) that is perpendicular to one of the given edges of the two polygons and when projecting all edges of the two polygons onto this axis results in no overlaps of the projected edges. [28]

For polygons it can be shown [28] that if two polygons are separated, the separation axis is always parallel to one of the polygons faces. In the case of squares this can be illustrated by the fact that all collisions happen either face to vertex or face to face. Here it is clear that as a collision approaches, the last separating line is the respective face.
This yields a robust algorithm for collision detection between two squares: If none of the faces of the two squares constitutes a separating axis, the squares are colliding. Computationally, this is done by projecting all vertices of the squares v_i onto the normals of the faces $\mathbf{n}_{\mathrm{AB}}$. A suitable choice for the projection operator is $\mathbf{P}_{\mathrm{AB}} = |\mathbf{n}_{\mathrm{AB}}\rangle \langle \mathbf{n}_{\mathrm{AB}}|$[2]. This approach projects the two-dimensional collision problem onto an one-dimensional collision problem, which can then easily be solved by comparing the maxima and minima of the two sets consisting of the vertices of the two squares. Since this is a somewhat geometric problem, Figure 3.2 is meant as a visual help for the explanation. The collision algorithm now checks all the faces of the involved squares as possible separating axes. If one of the faces tests according to case *a)* in Figure 3.2, a collision between the two squares can be ruled out.
An algorithm in this form would be quite expensive in respect to computation time, since for every positive collision 8 projections, extremals and comparisons would have to be computed. To reduce the computations needed, some properties of the problem are utilized.
First, for squares it always holds true that $\mathbf{n}_{\mathrm{AB}} = -\mathbf{n}_{\mathrm{CD}}$. But this implies $\mathbf{P}_{\mathrm{AB}} = \mathbf{P}_{\mathrm{CD}}$.

[2] Clearly, $\mathbf{P}_{\mathrm{AB}}$ fulfills $\mathbf{P}_{\mathrm{AB}} = \mathbf{P}_{\mathrm{AB}}^2 = \mathbf{P}_{\mathrm{AB}}^*$ and is therefore indeed an projection operator.

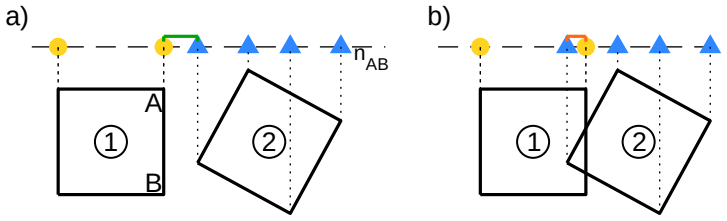

FIGURE 3.2: Detection of square collisions via SAT. The yellow circles represent the vertices of square **1** denoted as the set P_1. These are projected onto the normal n_{AB} of the face $\overline{\mathrm{AB}}$. The blue triangles represent the projections of square **2** P_2 . Since in *a)* $\max(P_1) < \min(P_2)$ holds true, the face $\overline{\mathrm{AB}}$ constitutes a separating line. In *b)* however, $\max(P_1) > \min(P_2)$ occurs. So face $\overline{\mathrm{AB}}$ does not constitute a separating line. These results can be easily verified by eye.

Therefore the projection onto the normals of two opposing faces yields the same result. This means that if one of two opposing faces is not a separating axis, the other one can not be a separating axis either. This alone cuts the computation time in half, since now only the two orthogonal faces of each square have to be checked.

Second, one can consider the inscribing and the inscribed circle in Figure 3.1. Two squares can not collide if the two circles inscribing the squares do not collide. Similarly, if the two circles inscribed by the squares collide, the squares must already be colliding. Detecting the collision of circles is computationally cheap compared to the collision detection of squares, since the collision of circles can be detected by simply looking at the distance between their centers r_{ab}. So a routine is added to the algorithm that only deploys the projection collision algorithm if $R_\mu + R_\nu < r < \sqrt{2}(R_\mu + R_\nu)$ and otherwise does or does not return a collision accordingly.

This algorithm can now successfully detect collisions between two squares. It is now manually tested for different cases of collision by checking the results of the algorithm by eye. It does not produce any wrong positives or negatives. It must be stressed that one has to especially mindful of the wrong positives, i.e. the algorithms test for a collision but no collision is present. This is because, in contrast to wrong negatives, wrong positives will not be detectable by just looking at the system states produced by the Monte-Carlo simulation after a run. Even if wrong positives occurred during the run, no overlaps would be present in the system. But the wrong positives could introduce a systematical error into the computation.

3.4 Cell Lists

The most straight forward way to determine whether a trial move generated in the Monte-Carlo algorithm should be accepted or discarded is to check if there are any collisions present in the system after the move is applied. Therefore the algorithm would have to check for any collisions between the displaced square and all other squares. Obviously, the computation time needed for this calculation scales linearly with the system size N. That in itself is not a bad characteristic, but if the scaling is to steep it can still be very difficult to achieve a high number of Monte-Carlo steps in large systems.
To alleviate this problem one can consider that, as mentioned above, the maximum distance between two interacting (colliding) squares is the sum of their inscribing circles $r_c = \sqrt{2}(R_\mu + R_\nu)$. This is the so called cutoff radius of the potential. If one would now divide the real space of the system into cells with edge length $r_c \approx L_c > r_c$, a square contained in a cell could only interact with squares in the same cell or in one of the eight neighboring cells. This reduces the number of necessary collision checks for big systems drastically, by even making it independent from the system size.[3] The corresponding cell list algorithm can be implemented into the Monte-Carlo algorithm in the following way [27].
Prior to the Monte-Carlo algorithm, the system is divided into square cells in such a way that there is an equal whole number of cells along both unit directions. Before generating a trial move, a cell list is created by assigning every cell all the squares contained within that cell. The data structure of the cell list is chosen as a linked list for reasons that will become apparent later. Now the trial move is generated and the cell containing the moved square is determined. After this, only collisions between the moved square and the squares contained in the cell of the moved square and its neighboring cells are calculated. If there is no collision detected here, the move is accepted. The legality of this procedure is ensured because collisions between the moved squares and squares not contained in the respective cells is impossible due to the ratio between cell lengths and the cutoff radius.
A further way to increase speed is the recycling of cell lists. If only one square is moved in each trial move, it seems excessive to recreate the cell list of the whole system before every move. And indeed, because of the linked list data structure, it is possible to efficiently replace just the moved square in the cell list and recycle the rest for the next trial move. This is done by comparing the cell of the moved particle before and after the trial move. If the cell number does not change, the cell list is simply passed on to the next trial move. If the cell number does change however, the index of the particle is removed from the linked list related to the old cell and simply added to the end of the

[3] Of course, the part of the algorithm that is responsible for accepting or rejecting trial moves still has a dependence on N, since the creation of the cell list must be included in it.

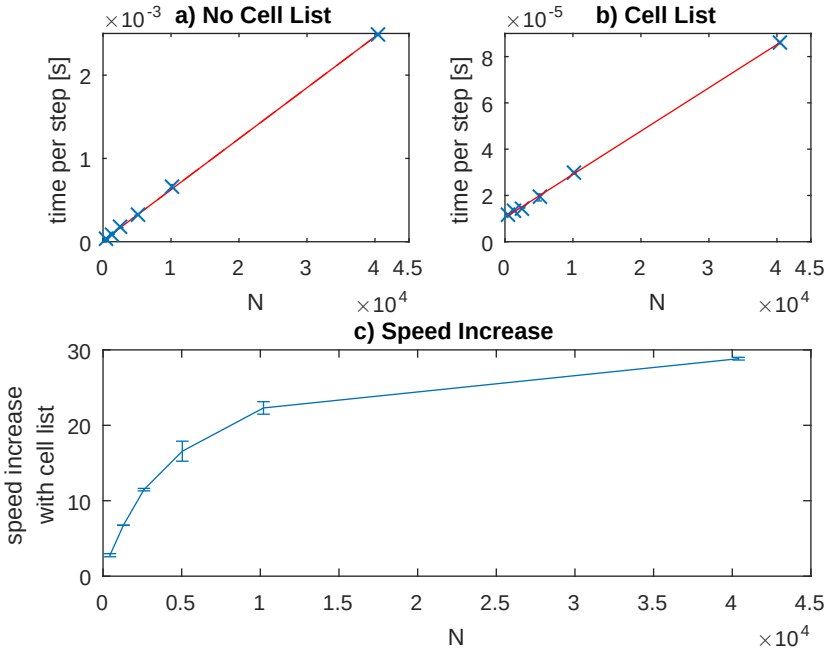

FIGURE 3.3: Speed increase of the Monte-Carlo algorithm by the implementation of cell lists. A system with packing fraction $\phi = 0.80$ is run for six different system sizes N. The CPU time per Monte-Carlo step without the implementation of cell lists is shown in panel **a)**. Panel **b)** displays the same time with the implementation of cell lists. The uncertainty is given by the standard deviation. The data is linearly fitted and yields a slope of $s_a = (6113.5 \pm 9.8) \cdot 10^{-11}$s for **a)** without cell lists and a slope of $s_b = (182.6 \pm 5.8) \cdot 10^{-11}$s for **b)** with cell lists. Panel **c)** shows the speed increase (time needed without cell lists over time needed with cell lists) over the system size N.

linked list related to the new cell.

The increase in speed with the cell list algorithm can been seen in Figure 3.3. It can be confirmed that the speed of the ordinary Monte-Carlo algorithm without cell lists does indeed scale linearly in N. Interestingly, the speed of the algorithm without cell lists does also scale linearly in N, although with a, by an order of magnitude, smaller slope. Since the number of squares checked for collision should not change with the system size, this scaling must be originating from another part of the algorithm. For the creation of the data in Figure 3.3 the Monte-Carlo algorithm is run in a core configuration with no routines apart from the time measurement and the simulation core itself. Therefore the linear scaling must be connected to some other internal part of the Monte-Carlo algorithm.

Comparing the two different computation speeds one finds that the algorithm with cell lists is only marginally faster for small system sizes and much faster for large ones. The ratio between the area of the system covered by nine cells and the whole area decreases with increasing system size, explaining this behavior. For the limes of infinitely large systems $N \to \infty$ the speed increase converges against the ratio of the slopes $\frac{s_a}{s_b} = 33.5 \pm 1.1$.

Since the Monte-Carlo algorithm will also be used for polydisperse squares, the im-

plications of this on cell list will be briefly discussed. To ensure proper results from the trial move acceptance algorithm, including the implementation of cell lists, the edge length of the cells should be equal or bigger than the maximum cutoff radius. This maximal cutoff radius is defined as $r_c^{\max} = \sqrt{2}\,[(R_\nu = \max_\mu(\{R_\mu\})) + (\max_\mu(\{R_\mu\}/R_\nu))]$. For systems with a high poly-dispersity this means that the cells will get significantly larger. Ergo more collisions become possible per trial move, which in turn slows down the Monte-Carlo simulation as polydispersity increases.

3.5 Initialization

Before the Monte-Carlo simulation can commence, the system must be initialized within the boundaries of the periodic box. In this thesis, two different approaches are chosen for initialization, depending on the polydispersity of the squares.

3.5.1 Initialization for Monodisperse Systems

Mono-disperse squares on a simple cubic (sc) lattice are space filling. This means that they can be placed on an *sc*-lattice while complying with a packing fraction $\phi = 1$. But this also means that only a few very specific ways are known to pack squares for high packing fractions [29]. Therefore, these systems are always initialized as *sc*-lattices.

3.5.2 Initialization for highly disperse systems

As will be shown in the following section, the initialization of polydisperse systems can be far more difficult. The problem presents itself as follows. As long as the polydispersity is low enough, the system can be initialized on a *sc*-lattice. But at a certain polydispersity p, the *sc*-lattice initialization could produce overlaps. This is the case when the maximum square radius R_m is larger than half of one of the sides of the unit cell. Since the unit cell length is always chosen as 1, and by regarding $R = \sqrt{\phi}/2$ and $R_m = (1+p)R$, one quickly finds that for initialization on a *sc*-lattice the polydispersity must fulfill

$$p < \frac{1}{\sqrt{\phi}} - 1.$$

So it is impossible to initialize high density highly polydisperse systems on a lattice. For these systems different approaches need to be found. Here, two different approaches are

employed.

The first approach has certain shortcomings and will only be described briefly. In this method, the system is initialized on a *sc*-lattice with a lattice constant large enough[4] to avoid any overlapping. The packing fraction is not yet equal to the target packing fraction as the box is too large. Now the squares are all sequentially moved in the x-direction until they hit another square or a wall. Here, the first problem of the method presents itself. Even though it would be in theory possible to predict for each square how far it could be displaced in one unit direction by again looking at projectors, the square is simply repeatedly displaced in that direction by a small amount until it detects a collision or hits a wall. This is very expensive computationally. After all squares have been displaced in the x-direction, the right wall of the periodic box is set to the farthest right square. This way all squares are still contained in the box, but the area of the box is reduced and the packing fraction is increased. Now all squares are displaced in the y-direction in the same manner and the area of the box is further reduced. Then the process is repeated for the $-x$ and the $-y$-direction. Eventually, all the squares fit inside a box that gives the desired packing fraction.

The major problem with this method is that it produces an initial state in which the squares are jammed. This means that many faces of the squares nearly touch. If one uses such a condition in a Monte-Carlo simulation, the jammed system leads to incredibly long equilibration times. So another algorithm has to be employed.

The idea for this algorithm is taken from [30]. Here, Lubachevsky and Stillinger are interested in densely packed spheres and describe an algorithm to achieve these dense packings. In their work they use molecular dynamics simulations and therefore the method described here deviates somewhat from the original. This idea is to let the system run in an ordinary fashion[5] but initialized with random positions and very small particle radii. Then the radii are uniformly expanded until the collision rate diverges. The adjustment for the method devised here are as follows.

The squares are still initialized on an sc-lattice. When the target radii of the squares in the poly-disperse states are denoted as $\{R_i^t\}$, then in the beginning all radii are chosen as $R_i = \langle R_i^t \rangle$. Then the Monte-Carlo algorithm from Section 3.2 is modified. After the random square is chosen, it is checked whether it has already reached it desired radius $R_i = R_i^t$. If this is true the algorithm, as usual, probabilisticly decides whether to perform a displacement or rotation move. If this is not true however, the algorithm adds a third possible move with probability 1/3. In this move, the square is neither displaced nor rotated, but the radius R_i is advanced to the target radius R_i^t by a step r uniformly distributed between in $[0, r_{\max}]$. Then the size adjusted square is checked for collisions

[4] It must fulfill $a = (1+p)\sqrt{\phi}$

[5] For Lubachevsky and Stillinger a normal molecular dynamics simulation for hard spheres, in this work a Monte-Carlo simulation as described above.

with other squares and if none occur, the move is accepted. Then a new random square is chosen and the algorithm continues to run until $R_i = R_i^t$ is true for all i. This method yields an initial condition that is far less jammed than that resulting from the previous method. For not too high densities and polydispersities, this algorithm is also a lot faster than the first method.

3.6 Proof of Equilibration

There is one more step required to ensure the validity of the data produced by the Monte-Carlo simulation. It should be tested whether the simulation can reach the equilibrium of the system.
In theory this must be possible, since the implementation of the algorithm after [6, 27] guarantees detailed balance and ergodicity. In practice the equilibration capabilities will be checked by two methods: equilibration from unfavorable initial conditions and convergence of Monte-Carlo step autocorrelation functions.

Monte-Carlo Step Autocorrelations The Monte-Carlo step autocorrelation is very similar to a time autocorrelation, only that Monte-Carlo simulations do not allow access to a physical time, but rather to a number of Monte-Carlo trial moves. With τ the amount of Monte-Carlo steps since some initial snapshot τ_0 and O some variable of the system, the Monte-Carlo steps autocorrelation is defined as

$$\langle O(\tau_0)O(\tau_0+\tau)\rangle(\tau) = \left\langle \frac{1}{N}\sum_{\mu=1}^{N} O^\mu(\tau_0)\left(O^\mu(\tau_0+\tau)\right)^* \right\rangle .$$

$\langle\rangle$ denotes the average over many snapshots. This function shows how much the configuration of particles at step τ_0 correlates to the configuration at step $(\tau_0+\tau)$. It is applied to all the local order parameters $O \in \xi, \psi_4, \chi$[6] and always shows the same behavior. The function drops quickly to a plateau and then stays there. The plateau is connected to the underlying order in the system. For example, in the solid phase a particle will rarely rotate more than a few degrees out of the plane due to long range orientational order. These functions will not be displayed here, as the snapshots were already chosen quite far apart and the autocorrelation functions already decay over two snapshots. But through comparison of all autocorrelation functions mentioned above in different phases, it is found that they all decay after 10^8 Monte-Carlo steps. In the remainder of the thesis, decorrelated snapshots will be referred to as snapshots that lie

[6] For their implementation, see Section 4.1.

at least 10^8 Monte-Carlo steps apart.
But how can this be used as a proof for equilibration? If one starts the autocorrelation in the initial condition, so $\tau_0 = 0$ and if the autocorrelation then decays to a plateau connected to the global symmetry, this means that the configuration of the system after the decay retains no memory of the initial condition. So it is independent from the initial condition. This is a very important criterion that must always be fulfilled in equilibrated systems.

Equilibration from Unfavorable Initial Conditions Another way to more understandably show the independence from the initial condition is to come up with extreme initial conditions and to see whether the system can recover from them.
Due to the high packing fractions, it is not trivial to choose different initial conditions than the *sc*-lattice in this system. Three approaches are chosen.

1. All particles are packed on a *sc*-lattice with a local packing fraction of one in one corner of the system.
2. The normal *sc*-lattice initialization is rotated by $\pi/4$.
3. After [29], a very extreme initial condition is chosen. It can be seen in the first picture in Figure 3.4.

Initial condition 1 quickly relaxes into configurations also found in systems initialized on a normal *sc*-lattice.
More problematic is initial condition 2. After approximately $2 \cdot 10^7$ Monte-Carlo steps per particle it shows only a very small rotation back to realignment with the periodic boundaries. It would probably take a very long time to reach configurations similar to those acquired from normal initialization. The question here is how much the properties of the rotated systems differ from the properties of the aligned systems. What can be said is that they are not exactly the same, since one lattice line would cross the periodic boundary many times in a rotated system, but only once in an aligned systems.
The equilibration of the third system is depicted in Figure 3.4. Here it can be seen that the initial condition first leads to the development of large coexisting regions with different particle orientations, which finally resolve to a system with one preferred particle orientation. However, this system is also slightly tilted in its global orientation.
From these equilibrations it can be inferred that the Monte-Carlo simulation is indeed able to equilibrate the system. The only problem here is the unknown influence of global system rotations on the behavior.

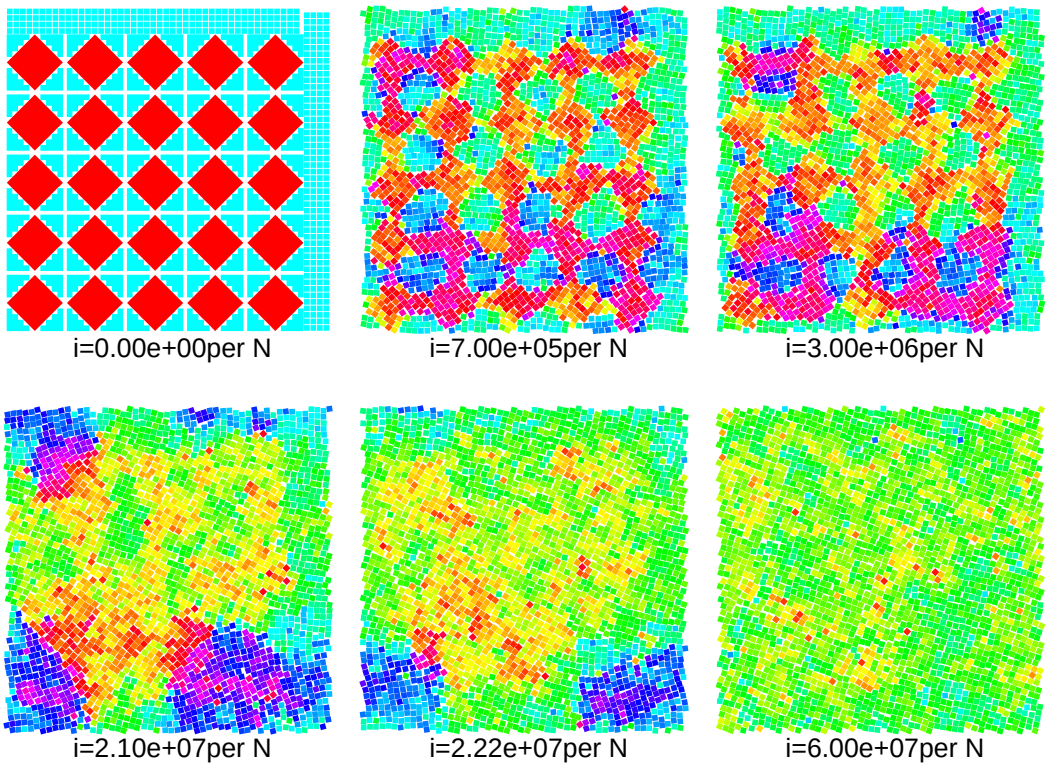

FIGURE 3.4: Equilibration from an extreme initial condition. Shown are snapshots taken at i Monte-Carlo steps per particle in a $N = 51^2$, $\phi = 0.80$ system. The colors of the particles represent the local body orientation order parameter ξ (see equation (2.3) and Section 4.1 for the implementation and mapping of color). This parameter visualizes the orientation of the particles, so similar colors correspond to similar particle orientations.

Chapter 4

Phase Determination

For later analysis it is vital to know the phase the system is in. Furthermore, phase transitions in two dimensions are very interesting, especially because of the intermediate *tetratic* phase between the solid and the liquid phases. The whole process of phase determination is further hindered by the fact that the points of the phase transitions depend on the system size. So a thorough phase analysis has to be done. This is achieved by first visually comparing the local order parameters and then looking at the scaling of order parameters in subsystems and extracting numerical values for the phase transition from this data. Finally the determined phases will be checked against the behavior of pair correlations of the same order parameters.

4.1 Implementation of the Local Order Parameters

In Section 2.1.2 the scaling of the local order parameters is connected to different phases. For this reason the local order parameters must be implemented first.

Local Body Orientation The implementation of this parameter (see equation (2.3)) is very straight forward, as the angle of the squares in respect to the y-axis is part of the simulation data.

Local Bond Order Nevertheless, the implementation of the local bond order (see equation (2.4)) has one subtle problem. This problem will be faced once more in the defect analysis in Section 5.2.1 and concerns the choice of next neighbors $\{\mathcal{N}\}$. The next neighbors can be identified in many different ways. The most popular ways are by Voronoi construction, by particles within a cutoff radius or by a chosen number of next

neighbors ordered after distance [31]. The last of these methods is selected here, because it is the only method that consistently produces the same number of next neighbors. For its implementation, for each particle all distances to all other particles have to be calculated and then sorted. This would be rather slow computationally and must therefore be optimized. One improvement is to only sort the distances inside a cutoff radius. The choice for the cutoff radius must be large enough so that always at least $\mathcal{N}$ particles reside inside of it. Another trick is to calculate the square of the distance and not the distance itself. This way the calculation of a square root can be omitted and since the square root is injective for positive real numbers the sequence resulting from the sorting does not change. All distances always occur twice per particle pair. Therefore even more computations can be left out when precalculating the distances so that each distance is only calculated once.
Now that the next neighbors are found, the calculation of the local bond order parameter can be easily done by formula (2.4).

Local Positional Order In order to obtain the local positional order parameter (see equation (2.5)), the first reciprocal lattice vector $\mathbf{q}_0$ has to be computed. This is done in the same fashion as is described in Section A. Even though the Bragg-peaks vanish for low packing fractions, the local positional order parameter shall still be calculated for continuity. As soon as the algorithm in Section A stops detecting Bragg-peaks, the reciprocal lattice vector is simply chosen as $\mathbf{q} = 2\pi\mathbf{e}_x$, as this is the reciprocal lattice vector of the perfect lattice.

Visual Representation of the Local Order Parameters As mentioned in the introduction to this section, the local order parameters shall first be compared by eye. So one has to find a suitable representation of the local order parameters. For this it is important to realize that all local order parameters are always inside the unit circle in the imaginary plane. So if a color wheel is placed inside the imaginary circle and if the particles are assigned colors corresponding to their position on the color wheel, the absolute value of the local order parameter will be represented by the saturation and the angle on the imaginary plane by the hue. A color wheel plotted on the imaginary plane can be seen in Figure 4.1.
When comparing different local order parameters it is helpful to multiply them with $e^{-i\varphi(\langle O\rangle)}$. Here $O \in \{\xi, \psi_4, \chi\}$, and φ is the angle of an imaginary number in polar representation. With this multiplication the average value of the local order parameter is always mapped the same color.

4.2 Visual Comparison of the Local Order Parameters

The local order parameters for each particle for different packing fractions are shown in Figure 4.1[1]. First, the behavior of the local order parameters shall be discussed separately. For ξ there is clearly a long range order for the packing fraction of $\phi = 0.85$, because the whole picture is of almost uniform color indicating that the values for ξ are uniform over the whole system. With decreasing packing fractions, the fluctuation of the values in the system increases, which can be seen in the more varied occurrence of colors. The point of the phase transition can not clearly be identified, but the nearly random fluctuations of colors for a packing fraction of $\phi = 0.70$ suggest that the range of the body order is short.
The local bond order parameter behaves in exactly the same way. Indeed if one looks closely, the colors in the ξ and the ψ_4 seem to be very similar, suggesting a strong correlation between the body orientation and the bond orientation. This correlation is most certainly evoked by the combination of relatively high packing fractions and the anisotropic potential. The high packing fractions force the particles to orient themselves after the anisotropy in the potential, leading to both structures in orientation of the particles and orientation of the neighbors.
A bit more revealing is the behavior of the local positional order parameter χ. For high packing fractions down to $\phi = 0.78$, the system exhibits differently colored stripes throughout the system. Again, since similar colors indicate similar values, this clearly indicates a local positional order, so a quasi-long range order. Below the packing fraction of $\phi = 0.78$, the colors suddenly start fluctuating wildly, indicating the loss of the quasi-long range order. So in between $\phi = 0.78$ and $\phi = 0.75$ a phase transition can be predicted.

4.3 Sub-Block Scaling

The Sub-block scaling analysis is a form of finite size analysis. What sets this method apart from other methods of phase determination is that it is computationally inexpensive. This advantage in speed originates from the fact that this method is based on a sub-division of the system into smaller sub-blocks. With Section 3.4, the algorithm for subdividing the systems into smaller cells is already implemented. This division scales with the number of particles in the system N, as the cell for each particle has to be identified. There remains only the task of averaging over particles, which also scales with roughly N in computation time. So the total scaling is approximately proportional

[1] If this is a digital copy of this thesis, feel free to zoom in on the system configurations. They are very interesting to look at microscopically.

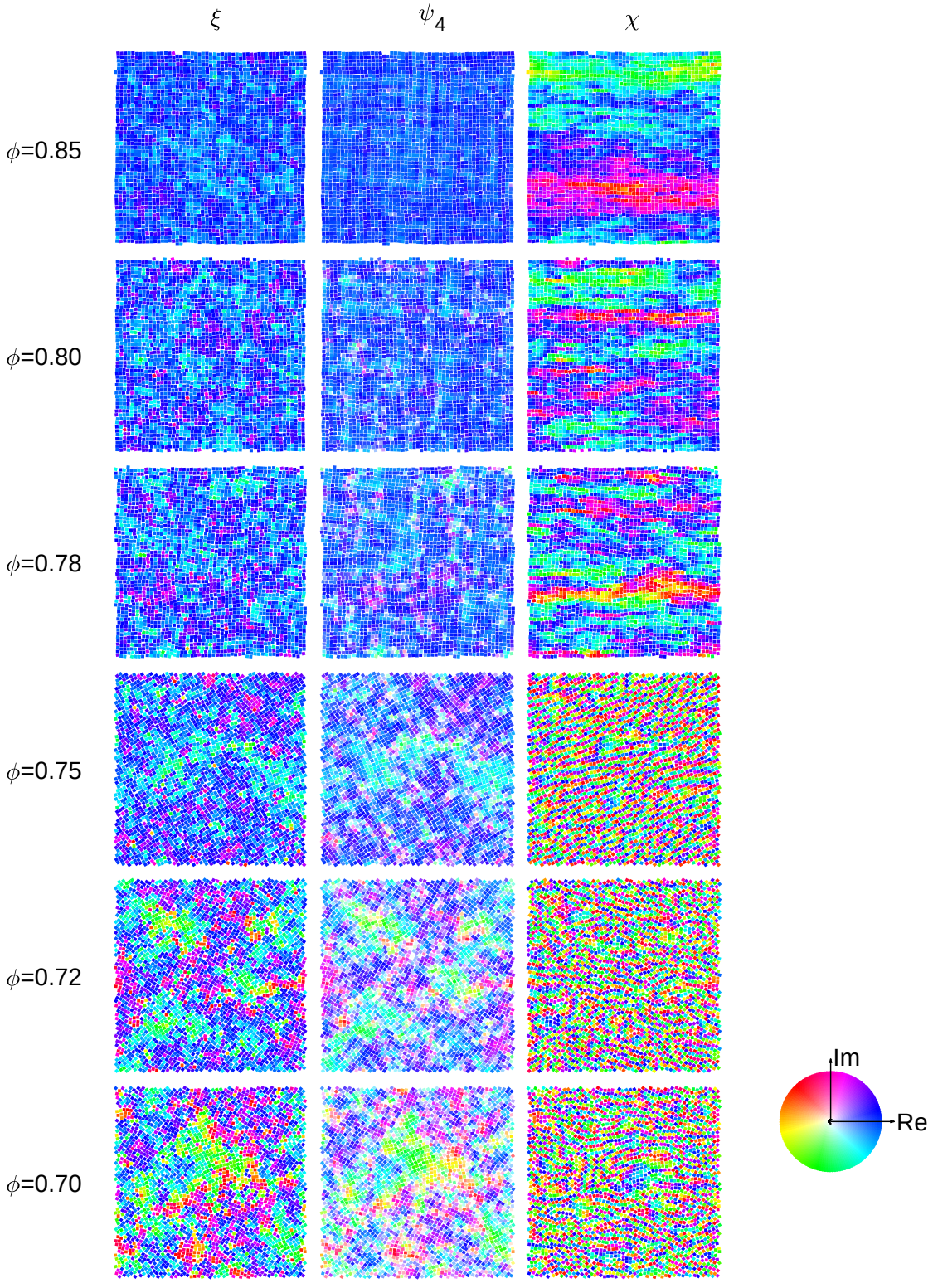

FIGURE 4.1: Visual comparison of the local order parameters indicating phase transitions. The local order parameters body orientation ξ, bond orientation ψ_4 and positional order χ are plotted for different packing fractions in a $N = 51^2$ system. The colors for the local order parameters are obtained as described in Section 4.1. With this, the color corresponds to the polar angle and the saturation corresponds to the absolute value. Therefore similar colors indicate similar local order parameters. Also shown is a color wheel visualizing the color mapping.

to N. Most other methods available look at some kind of *pair* correlator, indicating at least N^2 scaling of the computation time. Hence this method will be used to access numerical values for the phase transition.

Definition Here, the system of length L is subdivided into N_b square blocks of length L_b. Each of these blocks contains $N^i_{L_b}$ squares with indices I^i_b. Now the average order parameter ($O \in \{\chi, \psi_4\}$) per block is calculated through [1]

$$O(L_b) = \frac{1}{N_b} \sum_{\mu=1}^{N_b} \left| \frac{1}{N^\mu_{L_b}} \sum_{\nu \in I^\mu_b} O^\nu \right| . \tag{4.1}$$

According to the KTHNY-theory, in solids the positional order parameter χ decays by a power law with exponent $0 < t_\chi < \frac{1}{3}$ [8]. In the tetratic phase however, the bond orientation order parameter ψ_4 decays by a power law with exponent $0 < t_\psi < \frac{1}{4}$ [8]. Therefore one can determine the phase of the system by looking at the scaling of $O(L_b)$. The previous section has shown the strong correlation between ξ and ψ_4. It is therefore sufficient to analyze just ψ_4 with this method, as an analysis of ξ would only yield the same result as for ψ_4 again. To get better results and to introduce uncertainties in the form of standard deviations, $\langle O(L_b) \rangle$[2] is regarded. It is also easier to look at $\ln(\langle O(L_b) \rangle / \langle O(L) \rangle)$ over $\ln(L_b/L)$, because the power law exponents are equal to the slope of linear regressions in this representation. If one now plots the slopes of the linear fits in Figure 4.2 **a)** and **b)**, one can extrapolate the points of the phase transition from the points where the line connecting the slope values crosses the respective exponent. The uncertainty of these phase transition points are obtained in the following way. Respecting the standard deviation of $\langle O(L_b) \rangle$, the uncertainties of the slope of the linear regression are obtained. Then, the intersection of the uncertainty interval tube in Figure 4.2 **c)** and **d)** with the critical exponent is extrapolated. This intersecting interval is taken as the uncertainty of the phase transition packing fraction. As will be shown in Section 4.4, this method is too optimistic regarding the uncertainty of the phase transition.

The results from these calculations in form of phase diagrams will be shown in Section 4.4. Figure 4.2 shows an example for the sub-block scaling analysis evaluation of a $N = 51^2$ system. Here first some inadequacies in the application of this method shall be discussed.

When looking back at Tables 2.1 and 2.2 one can see that if all three phases, the solid, the tetratic and the fluid are to be expected, one should also expect constant, algebraic and exponential scaling in ψ_4 and algebraic and exponential scaling in χ. In the *ln-ln*-plot of this analysis method this translates to linear behavior for ϕ above the respective

[2] $\langle\rangle$ denotes the average over decorrelated snapshots.

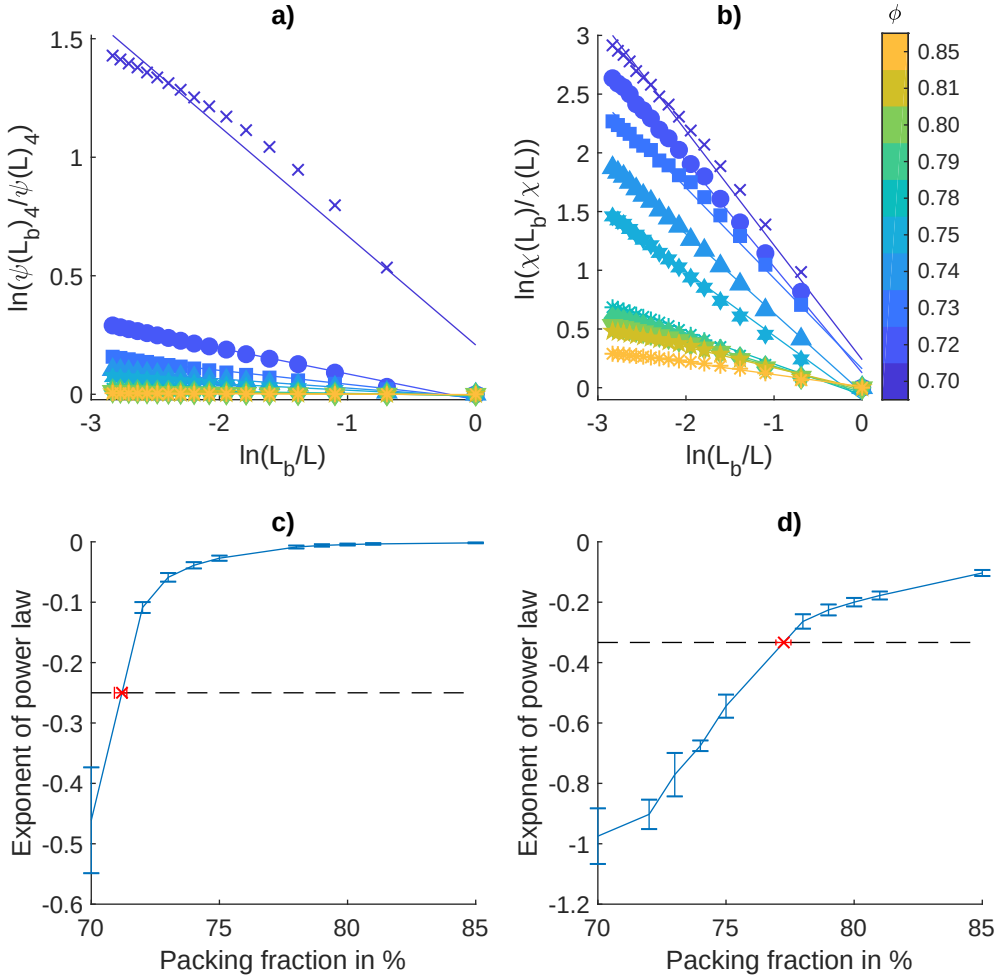

FIGURE 4.2: Sub-block scaling finite size analysis for $N = 51^2$ system. The figure is divided into four panels where the columns represent the analysis for the order parameter ψ_4 and χ respectively. The top row shows the sub-block scaling fitted with linear regressions. Averages of the local order parameters in the sub-blocks are calculated by equation (4.1). The bottom row shows the slope of those linear fits over the packing fraction. The critical exponents indicating phase transitions are plotted as dashed lines. Therefore the points of the phase transitions are given by the intersections of the slopes with the critical exponents. The acquisition of the uncertainties is described in Section 4.3.

phase transition and faster than linear decay for ϕ below the phase transition. And indeed this is very well visible in Figure 4.2. Especially for ψ_4 in panel **a)** it can be seen that the $\phi = 0.70$ curve decays faster than linear while all the other curves decay linear. Nevertheless the method described above explains that all curves are fitted linearly. This is not optimal, but the linear fits of the faster than linear decays will show to yield a sufficient approximation for the phase transitions. Note that the bottom row in Figure 4.2 shows the slopes of the linear fits in the top row with the uncertainties of these fits. These uncertainties also reflect the agreement of the data points with a linear curve. Inspecting panels **c)** and **d)** in Figure 4.2 one finds that indeed as soon as the slopes of the linear fits cross the critical exponent, the errorbars get much larger. This translates to bad agreement of the data from configurations below the phase transition with linear fits, which is exactly what Table 2.2 predicts. So the points of phase transitions found by this method agree well with the deviation from linear scaling in the *ln*-*ln*-plot.
It should be stated that the sub-block analysis method can also avoid these linear fits through exponential function by using data from systems of very finely screened packing fractions ϕ just around the phase transitions. But this takes many simulations and could not be accomplished in the time available in this thesis.
With the color coding of the curves in panels **a)** and **b)** in Figure 4.2 it can already be determined by eye that the slope of the linear fits generally decreases with decreasing packing fractions, indicating that the order of the system is decreasing with the packing fraction. This also confirms that the analysis will yield a unique point of phase transition.
Advancing to the bottom row **c)** and **d)**, the exponents of the linear fits plotted over the packing fraction show accordingly good results. The plot of the exponent verifies the monotonic increasing behavior and the extrapolated intersection with the critical exponents yields the packing fraction of the phase transition.
Unfortunately, the evaluation is not always as simple as in Figure 4.2. Sometimes the exponents of the linear fits are not strictly monotonic and cross the critical exponent several times. In this case the average of all the crossings is taken and the uncertainty is given by the lowest and highest limits of all involved uncertainty intervals.

4.4 Phase Diagrams

The finite size analysis outlined in the previous section is applied to all available systems from sizes $N = 21^2$ to $N = 101^2$ with monodisperse and polydisperse nature. The results are presented in the following section.

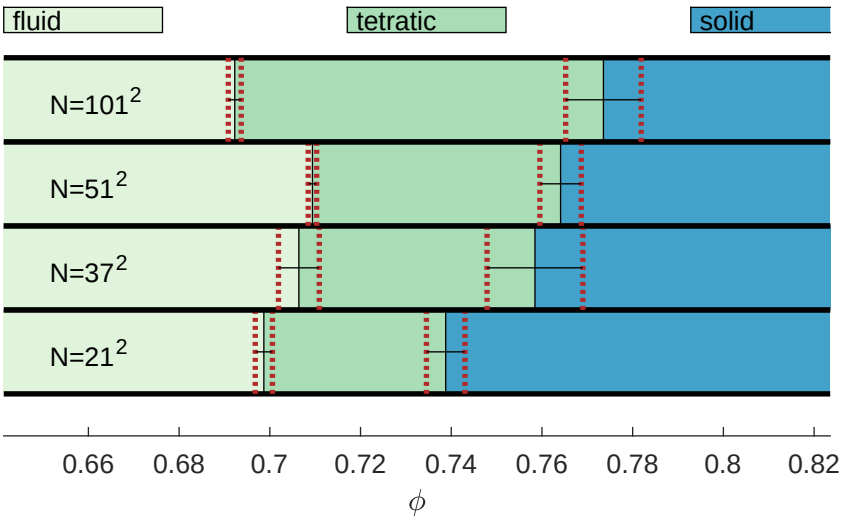

FIGURE 4.3: Tetratic phase appearing in phase diagram of monodisperse systems. In the phase diagram, the solid phase, the tetratic phase and the fluid phase are plotted over the packing fraction ϕ and for different system sizes N. This allows for a finite size analysis. The phase transitions are determined by a finite size analysis portrayed in Section 4.3 and their uncertainty is represented by the red dotted lines.

4.4.1 Phase Diagrams for Monodisperse Systems

The phase diagram obtained with the sub-block scaling finite size analysis is depicted in Figure 4.3. Three different phases occur in this phase diagram, the solid phase, the tetratic phase and the fluid phase. This is very interesting, since the tetratic phase predicted by the KTHNY theory indeed appears in this system.

It catches the eye that the phase diagram is only dependent on the packing fraction ϕ of the system. One usually sees phase diagrams with dependence on at least one other parameter. However, the system can be seen as operating in the microcanonical NVE-ensemble. Because of the hard core potential all allowed states have the same energy E and therefore the packing fraction ϕ is the only relevant parameter.

The next interesting thing is the scaling of the phase transitions with the system size. For the solid-tetratic transition this finite size analysis shows that the transition rises in packing fraction with increasing system size. For the largest system, the packing fraction for the solid-tetratic transition already climbed to $\phi_{st}^{N=101^2} = 77.36 \pm 0.83$. But a comparison with the findings of Glotzer *et al.* shows that the solid-tetratic transition will continue to rise with the system size to $\phi_{st}^{N=1024^2} \approx 0.84$ for a $N = 1024^2$ system [1]. Since there is no data available for the system sizes in between the $N = 101^2$ system simulated in this thesis and the $N = 1024^2$ system of Glotzer *et al.* it can not be said whether the solid-tetratic transition converges against $\phi_{st}^{N=1024^2}$ for $N \to \infty$ or if it continues to rise.

The finite size behavior of the tetratic-fluid transition is very different. There is no clear trend with system size detectable in Figure 4.3. The tetratic-fluid transition simply fluctuates around $\phi \approx 0.705$, but it is not constant within uncertainties. Anyhow, looking again at the results form Glotzer *et al.*, where the tetratic-fluid transition is found at $\phi_{tf}^{N=1024^2} \approx 0.71$ for $N = 1024^2$, suggests that the tetratic-fluid transition is indeed constant in the system size. This hints to the fact that the uncertainties for the transition are chosen much too optimistically. The reasons for this remain unknown.
But what does the behavior of the transitions in the system size reveal about the finite size effects on the order in the system? The rising of the solid-tetratic transition with system size implies that for smaller systems the solid, more ordered phase is adopted for smaller, generally more symmetric packing fractions. Therefore it can be deduced that the finite size of the system must induce some positional order to the system. If the tetratic-solid transition is considered to be constant with the system size this shows that the finite size of the system does not influence the rotational order in the system. This could be explained by the only indirect relationship between the orientation of single particles or neighbors and the position of the particles. Since the boundaries are implemented to be periodic they mainly influence the particle positions, as the rotation of the particles does not need to be aligned with the boundaries. It would be interesting to examine the same problem in a system with fixed boundaries, where the squares should orient themselves according to the boundary orientation. The here declared value for the tetratic-fluid phase transition in infinitely large systems should be compared with experimental data, as soon as it becomes available.
Lastly, the phase transitions for $N = 51^2$ compare well with the findings in Section 4.2.

4.4.2 Phase Diagram for Polydisperse Systems

The reasons why polydisperse systems are especially interesting for experimentalists is given in Section 3.1. But because they are interesting for experimentalists, it can be quite fruitful to perform phase analysis on these systems, as this could yield insights important for the setup of experiments. Luckily, the analysis method characterized in Section 4.3 is applicable to polydisperse systems without any changes being necessary. Figure 4.4 shows the phase diagram for polydisperse systems obtained with this method.

Here, the points of the phase transitions depend on the packing fraction and on the polydispersity, which is defined by $pd = \frac{\mathrm{var}(R)}{\langle R \rangle}$[3].
The solid-tetratic transition increases with increasing packing fraction. This effect gets quite severe for $pd = 12\%$, where the transition is already at $\phi_{st}^{\mathrm{pd}} = 86.4 \pm 1.9$. The

[3]Note that there is a difference between p defined in Section 3.5 and pd defined here. They are defined differently because p is easier to understand geometrically and pd is the value used in most literature. Since the radii are uniformly distributed, the conversion is $pd = p/2$.

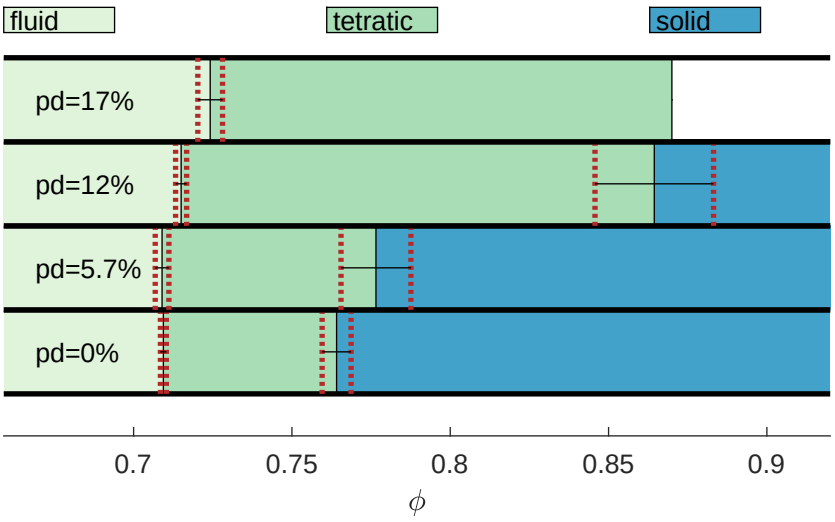

FIGURE 4.4: Stability of the tetratic phase in polydisperse systems. Shown is the phase diagram for $N = 51^2$ systems with different polydispersities pd. In the phase diagram, the three occurring phases, the solid phase, the tetratic phase and the fluid phase are plotted over the packing fraction ϕ and for increasing polydispersity pd. The rate of polydispersity is defined by $pd = \frac{\mathrm{var}(R)}{\langle R \rangle}$, with R the radii of the particles. The phase transitions are determined by a finite size analysis portrayed in Section 4.3 and their uncertainty is represented by the red dotted lines. For the highest polydispersity $pd = 17\%$, the solid phase could not be found in the accessible packing fractions.

case is even more extreme for $pd = 17\%$. Here, no solid phase is found in the interval $\phi = [0.68, 0.87]$ accessible in the simulations at hand. From this one can interpret that the polydispersity disturbs the positional order in the system. Geometrically, this can be explained by stating that it is impossible to position polydisperse squares with a higher polydispersity than $pd > \frac{1}{2\sqrt{\phi}} - \frac{1}{2}$ on a *sc*-lattice (see Section 3.5).

Much more interesting is the fact that the polydispersity only slightly influences the tetratic-fluid transition. It starts at $\phi_{tf} = 70.940 \pm 0.094$ for the monodisperse systems and only climbs to $\phi_{tf}^{pd} = 72.42 \pm 0.39$ for $pd = 17\%$ polydispersity. This indicates that the rotational order is not much influenced by the poldispersity. While it is not surprising that the squares in polydisperse systems still orient themselves in a preferred direction, implying long range or quasi-long range body order, it is somewhat surprising that the four-fold bond symmetry holds.

For experimentalists, this finding contains an interesting implication. It means that experimental systems can be initialized polydispersely and one can still expect to find the tetratic phase. Since some experimental colloidal realizations of the two-dimensional squares have some polydispersity in the particle size, and since finding the tetratic phase in a setup of that system is a validation the theory is still missing, this finding can indeed be useful and have application.

No finite size analysis for polydisperse systems will be presented here, as it yields exactly

the same results as the finite size analysis of the monodisperse phase diagram in Section 4.4.1.

4.5 Spatial Pair Correlation Functions

In the previous Section 4.3 the phase of the system was determined by looking at the scaling of order parameters in a finite size analysis. Another way to determine the phase is to look at the scaling of the order parameters by analyzing the behavior of spatial pair correlation functions. As already mentioned in 2.1.3, the solid phase is characterized by a quasi-long range positional and a long range rotational order, while the tetratic phase is characterized by short range positional order and quasi-long range rotational order. Lastly, in the fluid phase all order is short ranged. So one has to find a measure to compute the decay of order over distance. The tool of choice for this is the spatial pair correlation function C_O of the order parameter O.

Definition For the bond order ψ_4 and the body order ξ these correlation functions are easy to compute. Let $O \in \{\psi_4, \xi\}$. To calculate C_O, firstly for all pairs $\{\mu, \nu\}$ of squares $O^\mu (O^\nu)^*$ and the absolute distance between the squares $r_{\mu\nu} = |\mathbf{r}_\nu - \mathbf{r}_\mu|$ is calculated. Then the values $O^\mu (O^\nu)^*$ are binned according to the corresponding $r_{\mu\nu}$ with adequately chosen bin lengths Δ_b[4].
One problem in computation is that calculating this function for all pairs $\{\mu, \nu\}$ scales with N^2 in computation time and is therefore not possible for the larger systems. To solve the problem, one can choose p pair partners $\{\nu\}$ from all particles in the system at random without replacement. If there exists a certain global order in the system, this method captures that order, since it can be seen in the pairs chosen. If there exists only local order, the random choosing of the pairs makes sure that this order is averaged out, if not in one snapshot than at least in the averaging over multiple snapshots. So as long as p is sufficiently[5] large, this should not influence the result in a major way. Therefore the spatial correlation function is defined as

$$C_O(r) = \frac{\sum_{\mu=1}^{N} \sum_{\nu \in_p C_N} O^\mu \, (O^\nu)^* \sqcap_{\Delta_b} (r - \beta_{\Delta_b}(r_{\mu\nu}))}{\sum_{\mu=1}^{N} \sum_{\nu \in_p C_N} \sqcap_{\Delta_b} (r - \beta_{\Delta_b}(r_{\mu\nu}))}. \tag{4.2}$$

[4] The bins must be small enough to display the scaling of C_O but large enough to contain more than one value. Therefore the bin length strongly depends on the system size and must be evaluated for each system. It does not need to be exactly right, since as soon as a usable curve can be seen only its resolution not its behavior scale with the bin size.

[5] The size p does not influence the progression of the result, but the number of available data points and therefore the smoothness.

Here, the following definitions of $\sqcap_{\Delta_r}$ the rectangular function and β_{Δ_r} the binning function are used:

$$\begin{aligned} \sqcap_{\Delta_b}(a) &= \begin{cases} 0 & |a| > \Delta_b \\ 1 & |a| \leq \Delta_b \end{cases} \\ \beta_{\Delta_b}(a) &= \left\{ \left(c + \tfrac{1}{2}\right) \Delta_b \quad c\Delta_b \leq a < (c+1)\Delta_b. \right. \end{aligned}$$

Also, ${}_pC_N$ chooses p indices from N without replacement.
The calculation of the pair correlation for the positional order parameter χ is done differently. Here, the actual formula for the order parameter will not be used, because of its very strong dependence on $\mathbf{q}$ [1].The procedure is the following. [16]
All the distances between the squares $\mathbf{r}_{\mu\nu} = \mathbf{r}_\nu - \mathbf{r}_\mu$ are calculated and then binned in a two-dimensional bin grid. Here the bins are set fine enough, so that the highest peaks have a value of about ten. Now only bins lying close to the positive x-axis are considered (see Figure 4.5), yielding a one dimensional function. This function oscillates between zero and the maximal number of particles per bin for a certain distance. To observe the scaling of this positional spatial correlation function one is only interested in the peaks of this function. For large distances $|\mathbf{r}_{\mu\nu}|$ in x-direction, the choice of bins leads to there only being one or no particle in one bin. Therefore, the peaks converge to one. Since the scaling of this correlation function shall be observed in a 'log-log' plot, the value one is subtracted so that the function of peaks converges to zero. So,

$$C_\chi(r) = \sum_{\mu=1}^{N} \sum_{\nu \in {}_pC_N} \sqcap_{\Delta_b} \left(r - \beta_{\Delta_b}(r^x_{\mu\nu}) \right) - 1.$$

All functions $C_O(r)$ are averaged over 1000 decorrelated snapshots and the results for the body order and bond order pair correlation functions for a system of size $N = 51^2$ are depicted in Figure 4.6. The result for the positional order spatial correlation function is shown in Figure 4.7.

Results In Figure 4.6 the spatial correlation functions C_{ψ_4} and C_ξ are plotted over the spatial distance r. This is done in a *log-log* plot for a $N = 51^2$ system and for different packing fractions.
The first thing that catches the eye is the very similar behavior between C_{ψ_4} and C_ξ. This indicates a strong correlation between the orientation of the squares and the orientation of the neighbors of the squares, which has also been proposed in Section 4.2. Due to this similarity, the following discussion will only consider C_{ψ_4} in Figure 4.6 **a)**, as the discussion for C_χ in **b)** is analogue.
The average amplitude of one curve generally decreases with the packing fraction. Since

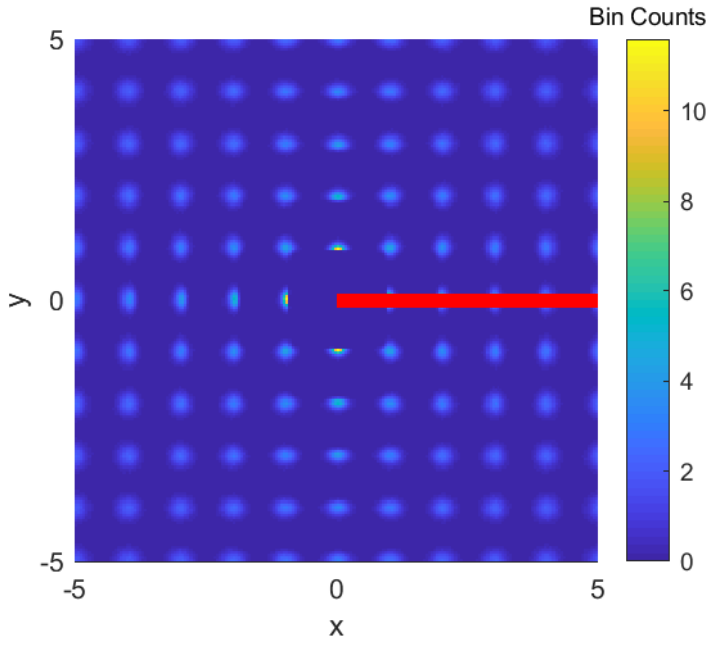

FIGURE 4.5: Indication of single phase behavior with two-dimensional pair correlation function $C_\chi(\mathbf{r}) + 1$. The values are obtained from a $N = 51^2$, $\phi = 0.85$ system (in the solid phase) by binning all distances between particles $\mathbf{r}_{\mu\nu}$ over a two-dimensional grid. A more yellow color indicates more particle distances in the system with the respective connecting vector. The red bar indicates the section from which values are taken to form the one-dimensional pair correlation function $C_\chi(r)$.

a lower spatial correlation connects to a lower spatial range of the order in the system, this demonstrates the global loss of order for decreasing packing fractions. As the packing fraction decreases, the system goes through the tetratic-fluid phase transition, where the KTHNY theory predicts loss of orientational order, so this observation is well compatible with the theory.

The amplitude also generally decreases with the distance r/σ, except for $\phi = 0.85$ packing fraction where it is almost constant. But more important here is the scaling of the correlation function with the distance.

For $\phi = 0.85$, $\phi = 0.78$ and $\phi = 0.75$ one can deduce that the bond order in the system is long range, since the scaling of the spatial correlation function converges on a constant value for large distances. For packing fraction $\phi = 0.72$ the decay of the spatial correlation function is not easy to classify for C_{ψ_4}. But since the strong correlation between ψ_4 and ξ is already established, one can also look at the $\psi = 0.72$ curve in the plot of C_ξ. Here, the long range behavior is clearly a linear decay in the *log-log* plot, meaning that it really has algebraic decay. This implies a quasi-long range bond and body order for $\phi = 0.72$. Lastly, the $\phi = 0.70$ curve decays faster than linearly in the *log-log* plot of C_{ψ_4}, indicating exponential decay and hence short range bond order.

This can be connected to the expectations of the KTHNY theory and the results from

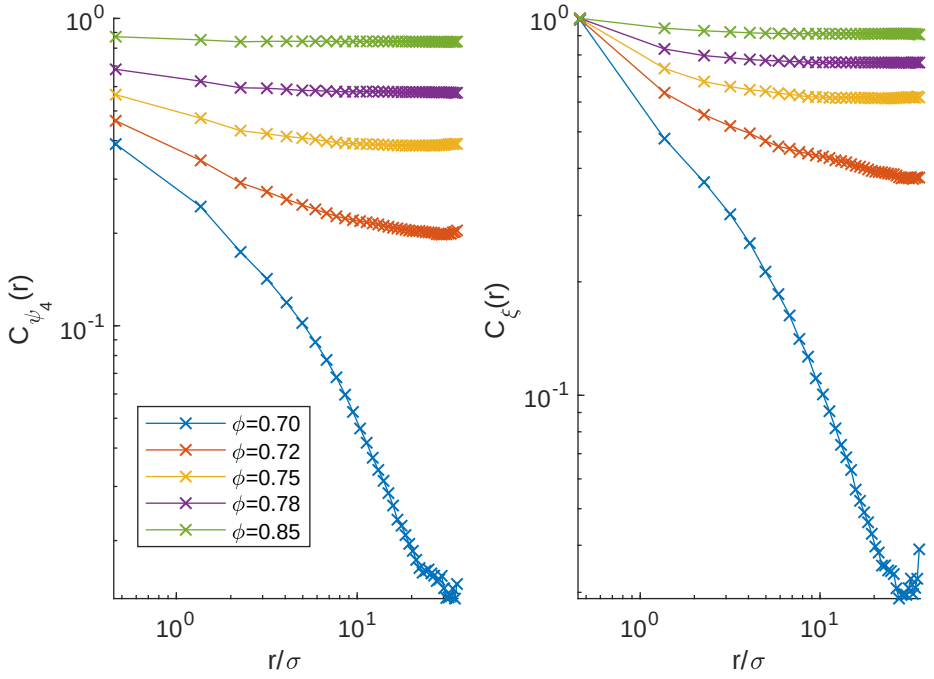

FIGURE 4.6: Angular spatial correlation functions confirming phase transitions. Shown are the spatial correlation functions C_{ψ_4} in **a)** and C_ξ in **b)** for different packing fractions ϕ in a $N = 51^2$ system. Both functions are plotted in a *log-log* plot over r/σ, where σ is the side length of the unit cell. The functions are calculated by equation (4.2) and are closely connected to the spatial scaling of the local order parameters. The data points are marked by the x and the connecting lines are meant to guide the eye.

Section 4.4. As already described in Section 2.4, the KTHNY theory links the melting of the system for the tetratic to the liquid phase to a loss in orientational order in the system. This means that at this melting transition, the order of the bond order parameter ψ_4 becomes short range. Since the only short range bond order is for packing fraction $\phi = 0.70$, the phase transition from tetratic to liquid must lie above this mark, but below the next long range packing fraction, so below $\phi = 0.72$. Comparing this with the results from the finite size analysis in Section 4.4, which states $\phi_{tf} = 70.940 \pm 0.094$ as the point of phase transition, gives good agreement.

Less compatible are the scalings for C_{ψ_4} found in the tetratic phase. The sub-block analysis predicts the tetratic phase to be between $\phi_{tf} = 70.940 \pm 0.094$ and $\phi_{st} = 76.41 \pm 0.46$. From Table 2.2 one would therefore expect quasi-long range order in ψ_4 for $\phi = 0.72$ and $\phi = 0.75$. As explained above, for $\phi = 0.72$ this behavior is found in C_{ψ_4}. However C_{ψ_4} also shows long range order in ψ_4 for $\phi = 0.75$. This is a discrepancy. The exact reason for this is unknown, but it could be connected to the continuous nature of the solid-tetratic phase transition. Considering this second order of the transition implies that the thermodynamical potential and therefore the order parameters are continuous in the first derivative and only discontinuous in the second, the order parameters will

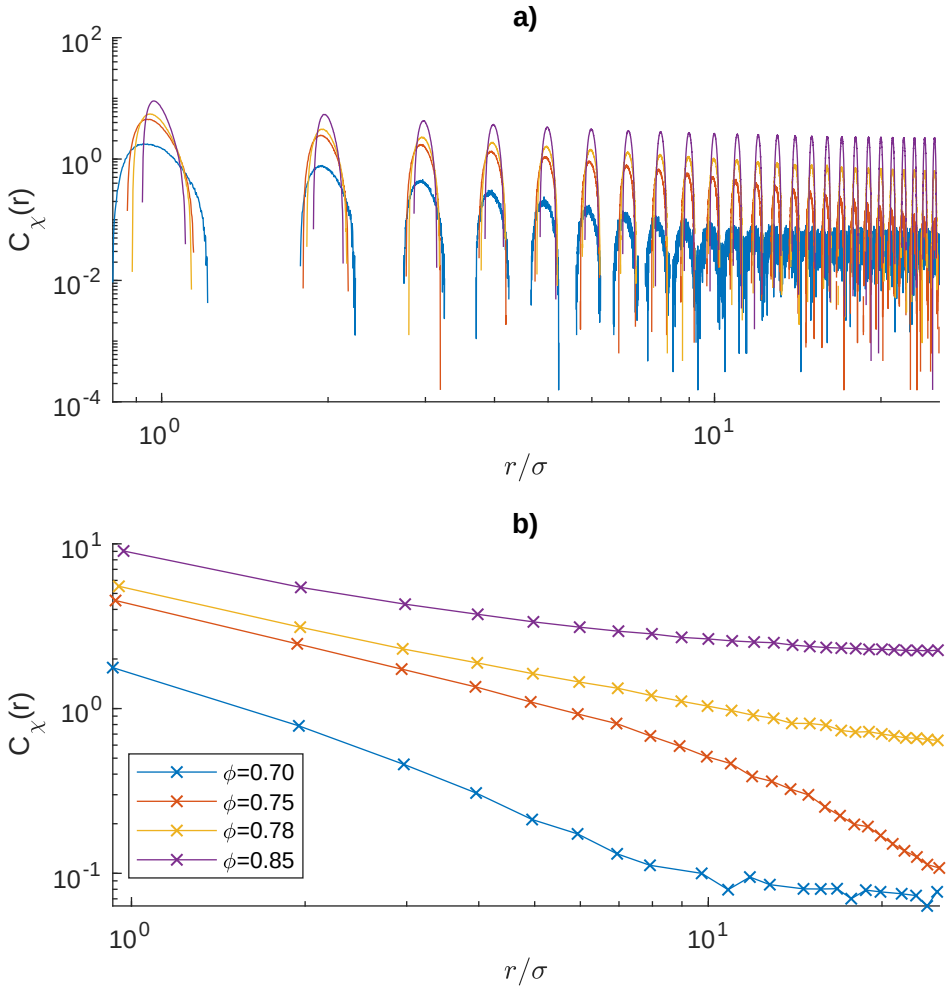

FIGURE 4.7: Positional spatial correlation functions confirming phase transitions.
a) The spatial pair correlation $C_\chi(r)$ plotted over spatial distances r/σ for different packing fractions. σ is the side length of the unit cell of a perfect *sc*-lattice corresponding to the respective system sizes. Note that this plot does not display the commonly seen x-axis of $r/2R$. $C_\chi(r)$ is calculated by binning distances between particles in the system, where the connecting vectors $\mathbf{r}_{\mu\nu}$ lie close to the positive x-axis.
b) The peaks of $C_\chi(r)$ plotted over r/σ for better visibility of the progression. The data points of the peaks are marked by the x and connected by a line meant to guide the eye.

not have a differentiable behavior in the phase transition, making them 'smooth'. But this also means that some behavior predicted in one phase can still show the effect in the other phase close to the phase transition. Crudely put, one could say that some of the long range bond order from the solid phase leaks into the tetratic phase. This discrepancy of the ψ_4 scaling can also be found in the figures produced by Glotzer *et al.* for the very large system of hard squares [32].
The $\phi = 0.78$ and $\phi = 0.85$ curves of C_{ψ_4} are placed in the solid phase by the sub-block analysis. Since they implicate long range bond order, this agrees well within Table 2.2.
For $r = 0$, Figure 4.6 shows the average of the autocorrelation of the respective order parameters in the system. The fact that this value is always equal to one for C_ξ but always smaller than one and decaying with packing fraction for C_{ψ_4} attracts attention. The behavior for $C_\xi(r = 0)$ can easily be explained. For $r = 0$ the values of the pair correlation are

$$\xi^\mu(\xi^\mu)^* = e^{i4\alpha_\mu} e^{-i4\alpha_\mu} = e^0 = 1.$$

For C_{ψ_4} the explanation is not so easy. It can easily be shown that the value must not be one, since the sum in the definition of ψ_4 keeps the conjugation from simply being the inverse as in the case of ξ. But it can be interpreted in the following way. $C_{\psi_4}(0)$ gives the squared amplitude of the order parameter ψ_4. As is explained in Section 2.1.2, this amplitude is connected to the accordance of the system with four fold symmetry. So the decrease of the autocorrelation implies some loss of the four fold symmetry.

The spatial correlation function $C_\chi(r)$ is displayed in Figure 4.7. The behavior of the peaks in panel **b)** for $\phi = 0.85$ and $\phi = 0.78$ is linear for long distances which implies quasi-long range order. This compares well to the fact that in Section 4.4 these packing fractions were identified as packing fractions belonging to the solid phase and that the KTHNY theory predicts quasi-long range positional order in the solid phase.
The curves for $\phi = 0.75$ and $\phi = 0.70$ decay faster than linearly for long distances and therefore imply short range order. This also compares well to the findings of Section 4.4, as these curves belong to the tetratic and fluid phase, where short range positional order is expected.
Also interesting are the positions and shapes of the peaks in Figure 4.7 **a)**. It shall be noted again that here C_χ is plotted over r/σ and not $r/2R$[6]. Two observations are to be made. First, the peaks grow wider with decreasing packing fractions and second the peaks shift inward with decreasing packing fractions. The maximum width of the peaks is given by the range of the hard core potential, since two particles can never move closer

[6]The plot over $r/2R$ is commonly found in literature. But the plot over r/σ, with σ the size of the perfect unit cell, can show the size change in the unit cell which will appear again in Section 5.1.3.

than this range. A reduction in packing fraction can also be facilitated by keeping the unit cell constant and reducing the size of the particles. But this does indeed reduce the range of the hard core potential allowing the particles to move closer, accounting for the increased width of the peaks.

The shift in the positions of the peaks is also very interesting. Since these positions dictate the distances between the average positions of the particles, the shift indicates the deformation of the unit cells. This effect will be discussed in more detail in Section 5.1.3.

Figure 4.5 can yield two more results connected to the solid phase in the system. Throughout this thesis it is always mentioned that the system crystallizes in a *sc*-lattice. This is clearly visible in Figure 4.5. Furthermore, in the thesis a phase is always declared as being purely solid or purely tetratic within the continuous nature of the phase transitions between all phases found in the system at hand. Figure 4.5 proves that the solid phase must be of single phase nature and does not have coexisting phases. If coexisting phases were present in the system, the peaks in Figure 4.5 would be much less clearly defined [16]. In order to finish up this discussion one last point will be raised. When looking at C_{ψ_4} and C_ξ in Figure 4.6 and at C_χ in Figure 4.7, it can be seen that the curves identified as scaling linearly, implicating algebraic decay, always slightly reduce their decay near the largest distances in the system. It is assumed that this is an effect stemming from the periodic boundary conditions, where multiple periodic images of particles correlate to other particles. This could be checked by comparing the curves in the figures mentioned above for different system sizes. Unfortunately this evaluation is not executed due to time constraints.

Chapter 5

Analysis of Crystallographic Defects

The analysis of defects in this system is very interesting. It is marked by the fact that all 'off the shelf' analysis methods for the local lattice structure fail in the system of hard squares. This is mainly due to the *sc*-structure. Therefore new methods for the detection of the local structure had to be devised. Using known methods as a basis many improvements where implemented, but most of them failed. Due to that, the nature of this chapter is somewhat exploratory and does not have a single objective. Maybe readers with similar problems can use this as a basis for further work.
First, the chapter will be dedicated to point defects and calculating a point defect density. Then it will outline a way to identify the local lattice structure. It was hoped that the local lattice structure in connection with the Burgers-vector could yield a topological defect density. Unfortunately, this could not be achieved in this thesis. Last, methods for identifying topological defects in the reciprocal space will be briefly examined.

5.1 Point Defects

To remind the reader, a point defect is defined as a vacancy, a missing particle from a lattice position, or an interstitial, an extra particle in a lattice position. First, an overview about special behavior concerning point defects in the system shall be given.

5.1.1 The Unlikely Occurence of Interstitials

The first specialty of this system is that it does only develop very few interstitials. Since an algorithm to microscopically find point defects could not be developed, this claim is

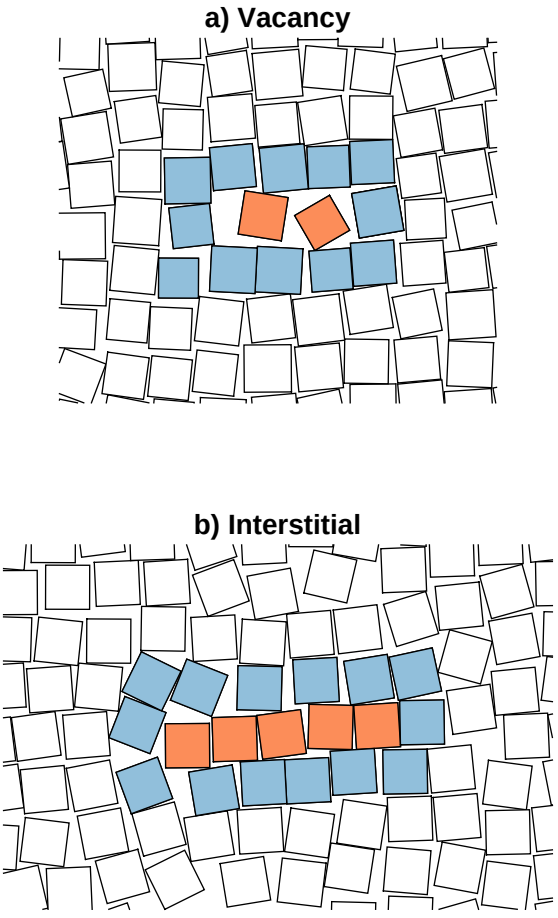

FIGURE 5.1: Delocalization of point defects in the system of hard squares.
a) A vacancy in a $N = 51^2$ and $\phi = 0.80$ system of hard squares. The vacancy is delocalized over three lattice positions. These three positions are only occupied by two particles plotted in orange. The particles surrounding the vacancy are plotted in blue and are relatively close to perfect *sc*-lattice structure.
b) An interstitial in a $N = 101^2$ and $\phi = 0.78$ system of hard squares. The interstitial is also delocalized by consisting of five interstitial particles (plotted in orange) distributed over four lattice positions. Notice how the interstitial distorts the surrounding lattice positions plotted in blue.

supported by three findings.

First, an extensive visual search for interstitials is done. Here, only very few defects could be clearly identified as interstitials. Such an interstitial is depicted in Figure 5.1.

Second, as the following Section 5.1.3 will show, the number of vacancies in the system is far greater than the number of interstitials. Such a strong dominance of vacancies could most likely not be achieved with many interstitials present in the lattice. Indeed for the hard core system of hard spheres, Frenkel and Pronk report three order of magnitudes less interstitials than vacancies [15]. A similar behavior is to be expected from

any hard core system.
Third, looking at a geometric construction can show the unlikeliness of the appearance of interstitials. It is already known that point defects create displacement fields in a crystal [33]. Since the system at hand is space filling, its solid phase occurs at comparably high packing fractions. This in turn leads to quite extreme displacement fields necessary in order to insert an interstitial. For the sake of simplicity, consider a system of perfectly aligned squares in a perfect *sc*-lattice. It can be easily seen that an inserted interstitial square creates the smallest displacement field if it is rotated by $\frac{\pi}{4}$. For a packing fraction of $\phi = 0.8$ one can easily compute that the particles surrounding the interstitial must be displaced by at least 0.4σ from their original lattice position in order to not overlap with the interstitial. This in turn leads to the fact that geometrically to avoid overlaps, the interstitials displacement field must stretch for at least four unit cells in each direction. Since this is only the geometrically possible minimum, it can be expected that these displacement fields are even larger in real systems. A vacancy however can be cured within a single unit cell, making them far more likely to be formed. This difference in displacement fields can also be seen in Figure 5.1. Observe how the particles surrounding the vacancies are quite close to the perfect lattice positions, while the lattice around the interstitial looks very distorted.

5.1.2 Delocalization of Point Defects

The second interesting feature of this system is the delocalization of point defects. Most vacancies in the system take the form shown in Figure 5.1. Here, the vacancy can not be mapped to a certain lattice position, but is delocalized between several particles, which in turn are surrounded by a box of squares sitting on their respective lattice positions. Looking at Figure 5.1, one can for example see that the vacancy is shared in such a way that two particles share three lattice positions.
Interestingly, this behavior has already been observed in the three-dimensional system of hard cubes by Dijkstra *et al.* [17]. For this system the vacancies also delocalize in one of the three unit directions of the system. Dijkstra *et al.* furthermore perform molecular dynamics simulations in non equilibrated systems, where the point defect density can be varied at will. From a maximum in the global order parameter over the defect density they deduce that a certain amount of point defects stabilize the system. Although similar behavior is to be expected for the two-dimensional system of hard squares, this can not be reproduced, since the implemented Monte-Carlo algorithm always yields equilibrated systems with non variable defect densities.
The less common interstitials also delocalize, as can be seen in Figure 5.1.

5.1.3 Point Defect Densities

The point defect densities found by Dijkstra *et al.* for three-dimensional hard cubes are as high as $n = 6.4\%$ [17]. To compare this to the two dimensional system at hand, the point defect density in this system shall be calculated.
If there are more vacancies than interstitials present in the system, the average number of particles per lattice site is smaller than one. But since the number of particles in the system N and the volume of the system V are fixed this means that the average volume of the unit cells must decrease. This would lead to there being more unit cells in the system, hence less particles per unit cell, ergo less particles per lattice position. Analogously, more interstitials than vacancies would lead to a increase in unit cell volume. An easy way to access the volume of the unit cell is through calculating the volume of the first Brillouin-zone. In the following the calculation for the point defect density will be shown.
The Fourier-transform gives the reciprocal lattice vectors $\mathbf{b_1}$ and $\mathbf{b_2}$. Now the basis vectors in real space are given by $a_i = \frac{2\pi \mathbf{b_i}}{b_i^2}$[1]. So the volume of the first unit cell is

$$V_U = |a_1 \times a_2| = \frac{(2\pi)^2}{b_1^2 b_2^2} |b_1 \times b_2| .$$

In the system of hard squares this volume is always smaller than the volume of the unit cell at initialization σ^2, so the number of units cells N_U is larger than the number of initialized unit cells N. Therefore, the number of vacancies in the system[2] is given by $N_V = N_U - N = \frac{V}{V_U} - N$. Lastly, the vacancy density n is given by

$$n = \frac{N_V}{N} = \frac{V}{N} \frac{b_1^2 b_2^2}{(2\pi)^2 |b_1 \times b_2|} - 1.$$

The results are averaged over 100 decorrelated snapshots and plotted in Figure 5.2. The uncertainties are obtained from the standard deviation. The vacancy density is plotted for different system sizes in order to include a finite size analysis. Here, not all packing fractions are represented in all system sizes. This is because firstly, the vacancy density is only defined in the solid phase and the larger systems melt for higher packing fractions. Accordingly, data for smaller packing fractions is only available for smaller system sizes. Secondly, for these smaller systems no simulations are done for higher packing fractions, since only a certain section of the solid phase is considered.
Three main observations can be found in Figure 5.2. The first one being that most

[1] This is only true since the reciprocal lattice vectors are orthogonal to one another because of the *sc*-structure of the system.

[2] This is not truly the number of vacancies, because an interstitial can cancel out a vacancy in this calculation. But since the number of interstitials in the system is low, this will be denominated as the number of vacancies.

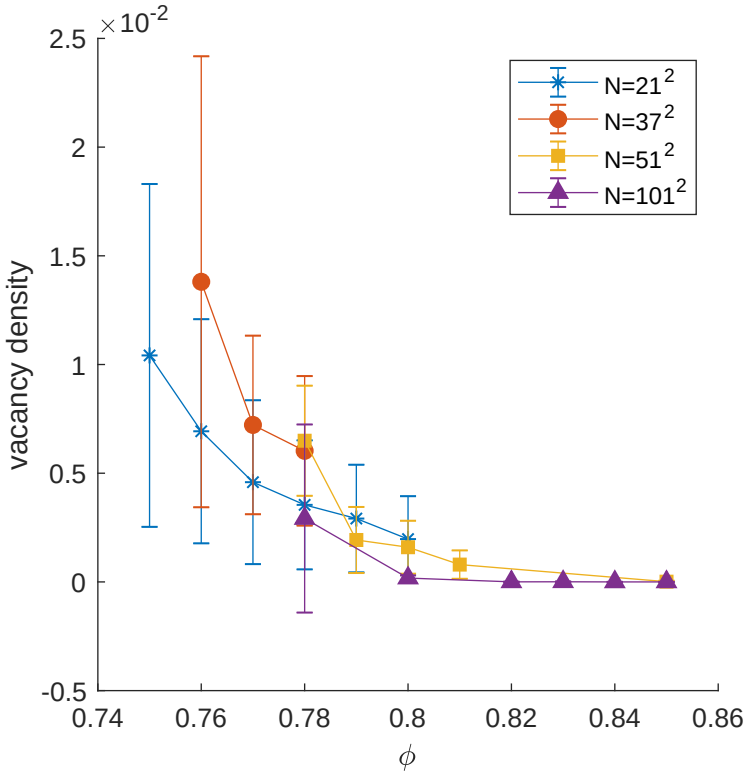

FIGURE 5.2: High vacancy densities in system of hard squares. Shown is the vacancy density plotted over the packing fraction for different system sizes N. It is averaged over 100 decorrelated snapshots and the uncertainty is the standard deviation. Note that a point defect density is only defined for systems in the solid phase. Because larger systems melt for higher packing fractions, the vacancy density for lower packing fraction is only defined for smaller systems. The highest occurring vacancy density in this graph is $(1.4 \pm 1.0)\%$ in a $N = 37^2$ and $\phi = 0.76$ system.

of the vacancy densities for different system sizes are compatible within uncertainties. Moreover, there is no clear trend for the behavior of the vacancy density with the system size[3]. Since Section 4.4 shows that the solid-tetratic phase transition is linked to the system size, and since the different systems sizes perform the solid to tetratic transitions at different vacancy densities it can be inferred that the appearance of point defects is not linked to the solid-tetratic phase transition of the system.

The second observation is that the vacancy density generally increases with decreasing packing fraction. This is to be expected, since smaller packing fractions make the system softer, which in turn favors the displacement fields introduced by point defects.

The third observation is the overall magnitude of the vacancy density. Vacancy densities of up to $n = (1.4 \pm 1.0)\%$ can be found. Compared with other two dimensional systems, this value is quite large. Zollweg *et al.* found point defect densities of $n \approx 2 \cdot 10^{-3}$ [34] for two-dimensional hard disk simulations, one magnitude lower than the defect densities found here. However, Dijkstra *et al.* found in their previously mentioned

[3]See Figure 5.2, packing fraction $\phi = 0.78$.

simulations of three-dimensional hard cubes point defect densities of up to $n \approx 6 \cdot 10^{-2}$ [17]. Since already the behavior of vacancies in the respective two-dimensional and three-dimensional systems of cubes is so similar, this similarity in high point defect densities is a further indicator for the tight connection between the behavior of point defect in both of these systems.
A problem that has to be addressed is the influence of the fixed periodic boundary conditions on the shape of the unit cell. Supposing that all locally defined unit cells in the systems are of roughly the same size, only discrete numbers and accordingly discrete sizes of unit cells are possible with fixed boundary conditions. Consequently the boundary of the system might influence the defect density extracted from the volume of the unit cell. Therefore the values for the defect densities obtained here might not be the values one would find in the real bulk of a system of hard squares. These influences of the static boundary could be eliminated by implementing fluctuating boundaries.

5.2 Topological Defects

The search for a reliable method for detecting topological defects within the system has proven to be very difficult. There are two basic ways to approach topological defects, in the real space and in the reciprocal space. Both of these possibilities will be described in the following section.

5.2.1 Topological Defect Analysis in Real Space

Approaching the topological defect analysis in real space, one has to know the local structure of the lattice. Then, a topological defect can be defined as a defect with a non zero Burgers vector. The local structure of the lattice is found through a modified version of the Voronoi tessellation, which will be described in the following.

Voronoi Construction with Face Weighting The local lattice structure can be found by looking at bonds between nearest neighbors in the system. Here, the Voronoi construction, or more precisely the Delaunay construction (see appendix B.2), is the tool of choice for most systems. But for this system, because it exhibits *sc*-structure, the Delaunay construction alone does not yield useful results. The reasons for this are closely connected to the explanation given in Section 5.3 about the instability of the Voronoi construction. If one wants to find an integration path for the Burgers-vector, it is important to go along the local *sc*-structure. The Voronoi construction however yields an average of six next neighbors, so there is also an average of six Delaunay bonds for

each particle. The task now is to decide which of these bonds belong to the local lattice structure and which can be ignored. To do so, for each particle the circumference L of the Voronoi cell is calculated. Then, the length of each Voronoi face L_i is determined. Each Delaunay bond is now assigned a weight equal to $\frac{L_i}{L}$ of the respective face.

Integration Path for Burgers-Vector The face weighted Delauny construction described above builds the basis for finding the correct integration paths for the Burgers-vector. The procedure will be presented in the following.
First, the number of steps to be taken in each direction is chosen. These directions are the global rotational orientation of the system and the orthogonal to this orientation. Since the integration path in an *sc*-lattice should be rectangular, the opposing sides of the path must consist of the same number of steps.
Then, a start particle is selected. The next particle position in the integration path is chosen by looking at all bonds found in the preceding paragraph that lie within a $\frac{\pi}{4} + \Delta$ angle to the first direction. If there is more than one bond within this angle, the bond with the bigger weight connects the integration path to the next particle. This procedure is repeated for the stated number of times in this first direction, and then accordingly in the second direction. To close the path the procedure is again repeated in the opposite first and second direction. As described in Section 2.3.2 the Burgers-vector is now given by the vector between the start particle of this construction and the end particle of the integration path. If the two particles coincide, the Burgers-vector is zero.

First Findings using the Local Lattice Structure The structure of the local lattice is given by the method outlined in the previous paragraphs. Due to time constraints, the following analysis is only performed in the solid phase of the system. This is quite unfortunate, because according to the KTHNY theory, the tetratic phase of this system should contain topological defects in form of disclinations. The structure of these disclinations in the system at hand it very interesting and not yet well observed [5].
First, the local lattice structure shall be presented visually, giving the reader a visual reference point for the following analysis. In Figure 5.3, the bonds obtained by triangulation are plotted.

The weights are represented by the opacity of the bonds. Bonds with large weights are black, while bonds with smaller weights are gray or even nearly white. Also plotted is a Burgers-vector integration path found with the algorithm presented in the previous paragraph. Since this path does not close, there is a Burgers-vector present and there must be topological defects within the box formed by the integration path. Finding that the Burgers-vector is also spanned by three lattice vectors, it can be inferred that there

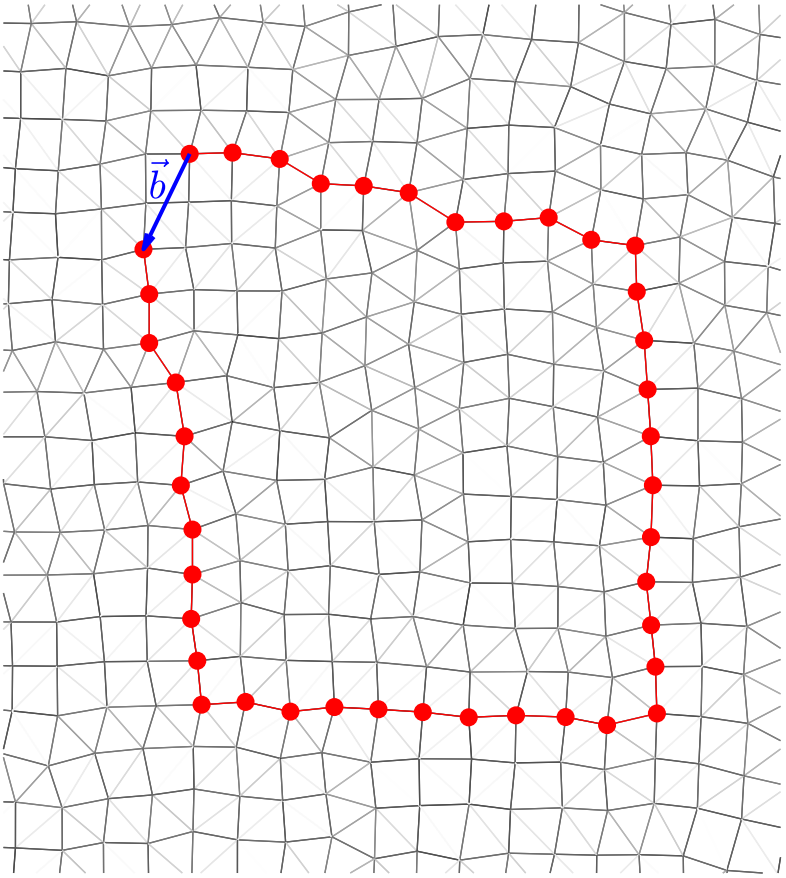

FIGURE 5.3: Defective local lattice structure and non-zero Burgers-vector. Shown is the local lattice structure of a $N = 51^2$ and $\phi = 0.78$ system. The local lattice structure is obtained by the algorithm described in Section 5.2.1. The bonds with large weights are represented by a large opacity and the bond with small weights with a small one. At each end point of a bond sits the center of a square in the real system. The squares themselves are omitted in this plot. The figure is meant to give the reader a visual basis for the analysis in Section 5.2.1. Also plotted is a non-closing Burgers-vector integration path (in red). This hints to topological defects being present within the integration loop. The Burgers-vector is printed in blue. Furthermore visible are the disturbances in the local lattice structure by the appearance of topological defects inside the integration loop.

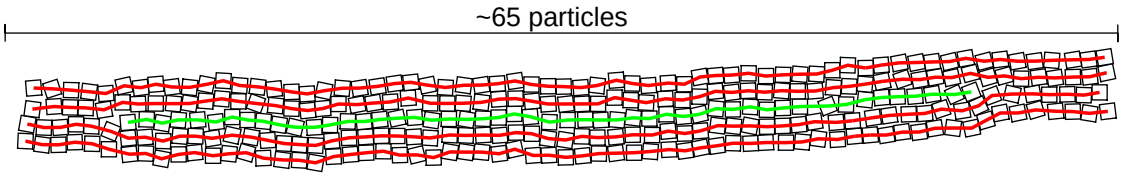

FIGURE 5.4: Inserted lattice line starting and ending in two dislocations. Depicted is a $N = 101^2$ and $\phi = 0.78$ system. The inserted lattice line is marked by a green line, the standard lattice lines of the *sc*-lattice are marked by red lines. The local structure of the lattice is found by the method presented in Section 5.2.1. Note that the inserted lattice line has a length of roughly 60 unit cells. Since the system is 101 unit cells long this means that the lattice line does not connect periodic images, but can be placed inside one box of periodic boundary conditions.

are three non canceling topological defects inside the box.
In the following some basic properties of topological defect in the solid phase of the hard square shall be discussed. A prediction that can be validated with the local lattice structure is the following. Because of the periodic boundary conditions, the system does not have a surface. If there is now a inserted lattice line starting at a dislocation in the system, it can not end in a surface, and therefore it should also end in a dislocation. Looking at many inserted lattice lines in the system has indeed shown this to be true. One of these inserted lattice lines is shown in Figure 5.4.

One could now think that if this figure can be created, than the problem of microscopically finding dislocations is solved. But this is not the case, because this figure is created by finding the dislocations by eye, and then letting the Burgers-vector integration path algorithm find the lattice line connecting the dislocations. The reason why this can not be automated is that the integration path algorithm will also connect the ends of the green lines in Figure 5.4 to neighboring particles, making it very difficult to pinpoint the location of the dislocation. The local lattice structure that presents itself around dislocations can also be seen in Figure 5.3. It distinguishes itself from the rest of the lattice structure by having lines of bonds not aligned with the global orientation of the system. The topological defects lie in these lines and create inserted lattice lines orthogonal to the lines of disordered bonds. For higher topological defect counts, one of these disturbed local structures can contain more than one topological defect, making the simple counting of these structures not a viable tool for topological defect detection.

Burgers-Vector Analysis The following data is extracted from the Burgers-vector integration path algorithm outlined in the previous paragraphs. Each particle in the system is taken as the center of an integration path. The side lengths of this integration path is varied between two and 50 unit cells by steps of two. Then the Burgers-vector and the norm of the Burgers-vector for this center and this side length is saved.

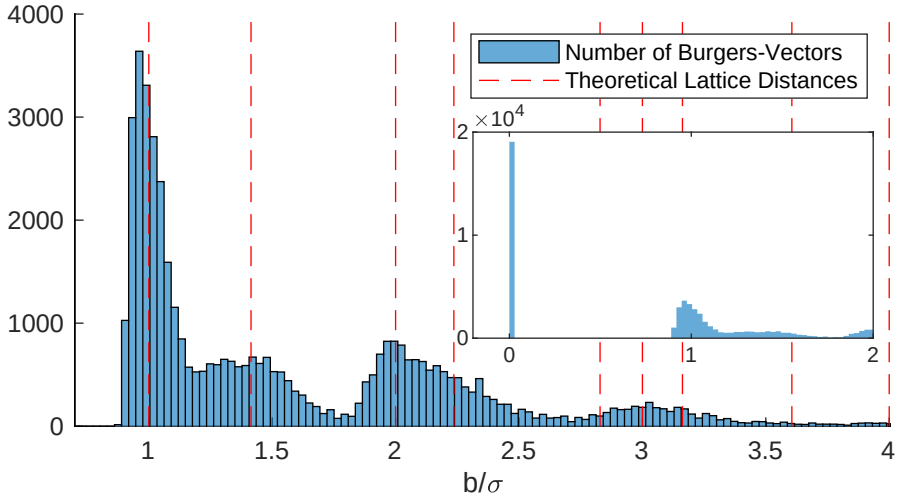

FIGURE 5.5: Occurring Burgers-vectors fitting on perfect lattice distances. Shown is a histogram of all Burgers-vector lengths b/σ occurring in a $N = 51^2$ and $\phi = 0.78$ system. σ is the side length of a unit cell in the perfect lattice. The highest peak at $b = 0$ is omitted in the main plot so that the other peaks remain visible. Its scale compared to the other peaks is therefore visible in the inset. The red dashed lines indicate the distances between lattice positions in the perfect lattice.

A histogram of all lengths of Burgers-vectors obtained this way is displayed in Figure 5.5.

In the histogram, the highest peak is located at $b = \|b\| = 0$. Therefore many integration paths do not yield a Burgers-vector. But should large integration paths not generally contain a high number of topological defects, creating the largest peak at large Burger-vectors? The answer is no, because if the integration path encloses two topological defects that would locally have opposing Burgers-vectors, then the two lattice lines inserted by the defects cancel each other out in the integration path and both topological defects give no contribution towards a Burgers-vector of this path. The number of Burgers-vectors per length decreases with increasing length, but some peaks are visible in this general trend. This general decrease is in accord with the argumentation above. The peaks can clearly be mapped to the distance between lattice positions in the system given by $d_l = \sigma\sqrt{a^2 + b^2}$ with $a, b \in \mathbb{N}$. The construction of the Burger-Vector in Section 2.3.2 indicates that the Burgers-vector can always be decomposed into lattice vectors, neglecting fluctuations of the lattice. This can indeed be confirmed by the preceding findings.

Due to time constrains, no further evaluation of the data is done. It should however be possible to calculate the number of topological defects from this data. For example, one could look at the number of integration path centers that yield a non-zero Burgers-vector with the smallest integration path circumference of four. Alternatively one could look

at the scaling of Burgers-vectors with the integration path circumference. With this it could even be possible to assign a size to each topological defect.

5.2.2 Topological Defect Analysis in Reciprocal Space

The topological defect analysis as previously performed in real space has a very tedious implementation, because the automated microscopic identification of the lattice structure is non trivial. As has already been done for the point defect analysis, a transformation to the reciprocal space promises to alleviate the trouble with the microscopic perception completely. Sadly though, other major problems replace the previous ones.

Basic Concept The basic concept of the influence of topological defects on the reciprocal space of a lattice is described in great detail in [33]. Here, only a very brief summary of these effects will be given in accordance.
The Bragg-peaks in the Fourier transformation of an infinitely large three dimensional crystal without any defects are given by delta distribution like peaks. If one now introduces some point defects to the system, the absolute amplitude of the Bragg-peaks decreases. An informal explanation for this can be given in the following. At not too high defect densities, all values of wave vectors that appear in the perfect crystal also appear in the defective crystal. This is because choosing one wave vector in the perfect crystal, one can always find a starting position in the real space of the defective crystal, where the Fourier transformed chosen wave vector lands on an occupied lattice position. This is true because the crystal is infinite. If one now introduces vacancies into the system, the total number of occurrences of one lattice distance in real space reduces, thus reducing the height of the respective Bragg-peak in reciprocal space. An interstitial however only adds one certain distance in real space, so one certain wave vector in reciprocal space. This occurrence is not periodic, so this wave vector will only give a small contribution to the background noise of the Fourier transformation. This in turn does also reduce the height of the Bragg-peaks. [33]

But how does a topological defect influence the reciprocal space? In Section 2.3.2 it is described how a topological defect influences particles at an arbitrary length from its origin. This also means that the defect perturbs a considerate part of the lattice structure. Take for example an inserted lattice line. Orthogonal to its orientation, it displaces all particles in the system, thus changing the distances between particles and with that the occurrence of certain wave vectors in reciprocal space. Indeed trough this influence, the Bragg-peaks of a crystal with topological defects loose their delta distribution like shape and now behave like Gaussian curves. Krivoglaz [33] even describes how in three

dimensions the density of topological defects can be extracted from the width of these Gaussians.
Because of deficits in the definition of topological defects via the Burgers-vector explained in Section 2.3.2, it would be best practice to first analyze topological defects in reciprocal space and then compare the results from this analysis with results from an analysis in real space. Unfortunately, the analysis in reciprocal space could not be completed in this thesis.

Problems The configuration of the system at hand introduces some difficulties in using this analysis method.
Firstly, in an infinitely large two dimensional system there exist no Bragg-peaks, because the positional order is faulted by Mermin-Wagner fluctuations and therefore only has quasi-long range order. Accordingly the concept illustrated above is not applicable for these systems.
Luckily the finite size of the system at hand reintroduces the Bragg-peaks to the reciprocal space. But there exists no theory[4] linking the width of Bragg-peaks in finite two-dimensional systems to the appearance of topological defect.
Another problem is given by the periodic boundary conditions. As stated in Section 2.6.6, they limit the number of possible wave vectors in the system, so that these vectors only appear at discrete values. If now the Bragg-peaks change width due to the introduction of topological defects, but the change in width is smaller than the distance between two discrete wave vectors, it is effectively not detectable. As already mentioned, Krivoglaz connects the width of the Bragg-peaks to the topological defect density, hence to the distance between topological defects. In a system without periodic boundary conditions, the distance between the topological defects is also the length of the inserted lattice line. In a system with periodic boundary conditions however, the distance between the topological defects does not need to be the length of the inserted lattice line, since the lattice line can connect a topological defect to a periodic image of a topological defect. This means an inserted lattice line can cross the periodic boundaries multiple times, which is indeed being observed in the simulations at hand. The question now arises which distance is important for the definition of topological defect densities and which distance corresponds to the width of the Bragg-peaks. A definitive answer to this has not been found, but since the inserted lattice lines in the systems at hand tend to be longer than the length of the system, the reciprocal analysis will most likely not be meaningful. The solution to this problem is simulating larger systems in order to increase the number of discrete wave vectors and decrease the chance of a lattice line

[4] To my knowledge.

crossing the periodic boundaries more than once. Due to constraints in the computation time however, the implementation of this solution is not possible in this thesis.

5.3 Insufficient Analysis Methods for the Local Lattice Structure

As previously mentioned, the implementation of reliable defect analysis code for the system at hand was a lengthy process of failing algorithms. In order to save others from using these insufficient methods, they will be presented in this section, together with their specific reason of failure.

The Regular Voronoi Construction Topological defect analysis by Voronoi tessellation is a widely used method in the field [1, 13, 31]. The basic principle is to use the Voronoi construction to gather the number of next neighbors for each particle. Two points are identified as next neighbors, if their respective Voronoi cells share a face. This number of next neighbors is six on average, and each particle deviating from this average must be a point defect or must lie next to a point defect. For the monoclinic, centered orthorombic and hexagonal lattices, this approach yields very good results. For all other Bravais-lattices however it has great deficits, which will be explained by looking at the tetragonal lattice. In a perfect *sc*-lattice, the Voronoi tessellation yields four next neighbors for a particle, as is to be expected. But now consider a thermally fluctuating lattice, as is shown in Figure 5.6. Configuration **a)** and configuration **b)** are very similar, but through the Voronoi construction the Delaunay triangulation yields a neighbor count of eight for configuration **a)** and a neighbor count of four for configuration **b)** for the central particle, but neither configurations depict or are close to a point defect. In fact, this regular Voronoi construction applied to a fluctuating *sc*-lattice just yields randomly fluctuating values for the number of next neighbors. It is therefore in itself not a suitable tool for point defect analysis in this system.

The Voronoi Construction with Clustering Consider again Figure 5.6. If the number of next neighbors for the central particle increases as in configuration **b)**, then the number of next neighbors for some surrounding particles decreases. Because of this relation, an algorithm is implemented that maps all defects to clusters of defects. These clusters are defined as all defects that can be connected by next neighbor relations, whereas defects are defined as particles with a non six neighbor count. The idea is that defects receiving a next neighbor bond through fluctuation cancel out with defects in the crystal not receiving these extra bonds.

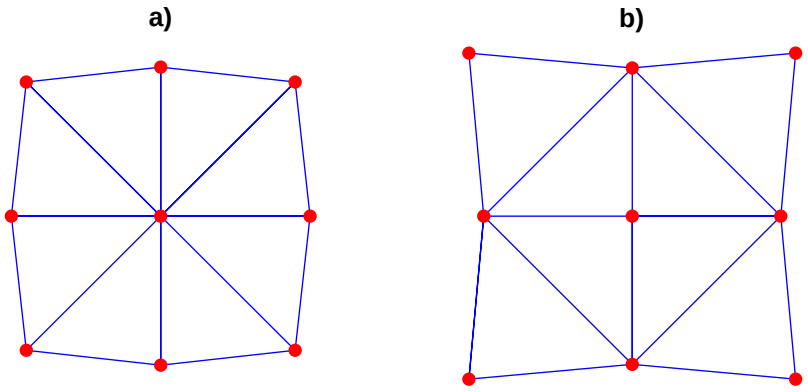

FIGURE 5.6: Similar configurations yielding different Delaunay bonds. The red dots represent the particle positions. The blue lines are the Delaunay bonds. Note how these similar configurations yield different neighbor counts for the central particle, namely eight in **a)** and four in **b)**.

This approach has one inherent flaw. The density of defect defined this way is so high that a defect cluster can easily stretch to length scales of the whole system. Accordingly, the defect count in these clusters averages out to zero, independent of the identification described earlier. So this approach is again not viable.

Common Neighbor Analysis Another tool suitable for structure identification is the common neighbor analysis, as described in [35]. Here, the next neighbors of a origin particle are identified via the Voronoi construction and then the next neighbors of these next neighbors are again identified by Voronoi construction. This yields a signature of the number of neighbors shared by the origin particles and the next neighbors and the number of neighbors shared between the neighbors of the origin particle. In three dimensions this signature is unique for different underlying lattice structures and can, if this lattice structure is already known, be used to identify defects by deviations from the expected signature.
This method is implemented but yields no improvement over the regular Voronoi construction, as it exhibits the same problems.

Bond Length One approach to detecting point defects microscopically is to consider bond lengths in the system. Since point defects are disturbances in the local lattice structure, the length of bonds connecting the lattice points varies around a point defect. In order to use this information, for each particle the next four neighbors are calculated

using a Verlet list. This means that the absolute quadratic[5] distance between the particle and all other particles are calculated and then indices of particles with a distance under a certain threshold are saved. For these saved particles the ones with the four lowest distances are selected.
For these neighboring particles the distance is averaged in order to give the average bond length of the particle. The problem in evaluating this data is that one would have to proceed with a parameter dependence. One would have to decide for which deviation from the average bond length a particle is counted as a defect. And even if one decides on such a parameter, it still should not be evaluated as a simple threshold, because this would again give rise to random noise due to fluctuation. All this makes this a very subjective approach, and it is therefore discarded.

Point Defect Analysis in Real Space Using the face weighted Delaunay construction, it is tried to microscopically search for point defects by again looking at the number of next neighbors. This number is obtained by only counting particles as next neighbors, when the weight of the bond is over a certain threshold Δ_T. Since a four fold symmetry in the neighbor structure is suspected, $1/4 \lesssim \Delta_T$ is chosen. The problem here is again the parameter dependence. At the time this method is implemented no reference data existed to estimate this parameter. But since now the results of Section 5.1.3 are available it would be interesting to check whether it is possible to calculate a suitable estimate for Δ_T.

[5] Not calculating the square root saves computation time.

Chapter 6

Dispersion Relation

With the previous sections, the groundwork for the dispersion relation is done. Now, the latter shall be obtained. This is done by calculating the eigenvalues of the matrix $\mathbf{\Lambda}$ defined in Section 2.5.2. Therefore the following section concerns itself with the computation of this matrix.

6.1 Retrieval of the Λ-matrix

As a reminder, the $\mathbf{\Lambda}$-matrix is defined as

$$\Lambda^{\alpha\beta}(\mathbf{q}) = Vk_BT\sum_{\mathbf{g}'\mathbf{g}}(g'+q)^{\alpha}n^{*}_{\mathbf{g}'}\left(\left\langle\delta\rho^{*}(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\right\rangle^{-1}\right)^{*}n_{\mathbf{g}}(g+q)^{\beta}. \tag{6.1}$$

So first, the average height of the Bragg-peaks $n_{\mathbf{g}}$ needs to be calculated through

$$n_{\mathbf{g}} = \frac{1}{V}\left\langle\sum_{\mu=1}^{N}e^{-i\mathbf{g}\cdot\mathbf{x}_{\mu}}\right\rangle.$$

$\langle\rangle$ denotes the average over different snapshots of the system. Here, the angular coordinates have to be taken into account, so the following is a reminder for all definitions of coordinates. $\mathbf{r}_{\mu}$ are the particle positions and α_{μ} are the angles of the particles in respect to the y-axis. Then, $\mathbf{x}_{\mu}$ is defined as

$$\mathbf{x}_{\mu} = \begin{pmatrix}\mathbf{r}_{\mu}\\ \alpha_{\mu}\end{pmatrix}.$$

$\tilde{\mathbf{g}}$ are reciprocal lattice vectors and m are the Fourier series components of the angles, hence $m = 0, 4, -4, \ldots$. Similarly, $\mathbf{g}$ is defined as

$$\mathbf{g} = \begin{pmatrix} \tilde{\mathbf{g}} \\ m \end{pmatrix}.$$

So $n_{\mathbf{g}}$ reflects both the order in the position and the order in the orientation. It becomes apparent that one has to choose which $\mathbf{g}$ to take for the calculation of $n_{\mathbf{g}}$ [1]. Theoretically, there are as many unique $\mathbf{g}$ as there are Bragg-peaks in the three-dimensional reciprocal space. But the following analysis algorithms computation time will scale heavily[2] with the number of $\mathbf{g}$. So, the $\mathbf{g}$ will be chosen by their neighbor generation. In the reciprocal *sc*-lattice the first generation of neighbors contains eight $\tilde{\mathbf{g}}$ and the constant peak $n_{\mathbf{0}}$. The second generation contains the previous nine peaks plus 16 more and so on. But in the Fourier series, each generation of neighbors contains two more m. So the first generation is $m = \{0, -4, 4\}$, the second $m = \{0, -4, 4, -8, 8\}$, etc.. Therefore the first neighbor generation of $\mathbf{g}$ contains $9 \cdot 3 = 27$ vectors, the second generation contains $25 \cdot 5 = 125$ and so on. Because the number of vectors grows so quickly, only first and second generations will be regarded.

In order to now calculate $n_{\mathbf{g}}$, many decorrelated snapshots of the system are prepared for analysis using the method from Section A. For each of these snapshots, the $n_{\mathbf{g}}$ for the appropriate number of neighbor generations are calculated and then averaged over all snapshots. The matlab code is vectorized for speed increase (see Section D).

As a next step, the reciprocal density fluctuation correlation matrix $\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\rho(\mathbf{g}'+\mathbf{q})\rangle$ needs to be calculated. It is given by [36]

$$\left\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\right\rangle = \left\langle \rho^*(\mathbf{g}+\mathbf{q})\rho(\mathbf{g}'+\mathbf{q})\right\rangle - \delta_{\mathbf{q},\mathbf{0}} V^2 n_{\mathbf{g}} n^*_{\mathbf{g}'}.$$

Inserting the reciprocal density (2.8) yields

$$\left\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\right\rangle = \left\langle \sum_{\mu,\nu=1}^{N} e^{i\mathbf{g}\cdot\mathbf{x}_\mu} e^{-i\mathbf{g}'\cdot\mathbf{x}_\nu} e^{-i\mathbf{q}(\mathbf{r}_\nu - \mathbf{r}_\mu)} \right\rangle - \delta_{\mathbf{q},\mathbf{0}} V^2 n_{\mathbf{g}} n^*_{\mathbf{g}'},$$

because q is always defined as

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ 0 \end{pmatrix}.$$

[1] The same choice must be made for the calculation of the whole $\mathbf{\Lambda}$-matrix.
[2] No exact number is computed, but approximately with $N_{\mathbf{g}}^4$.

$\mathbf{q}$ should be chosen along one of the symmetry directions shown in the inset of Figure 6.1. It is first tried to compute the expression for $\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\rangle$ with a highly vectorized code. But even with the vectorization, the code is much too slow, potentially needing weeks of calculation time to yield results.
The method that is being used for calculation was suggested and implemented by Johannes Häring. It is much faster, reducing calculation times to only a few minutes. The basic idea is to enable the use of a fast Fourier transform (FFT) by binning the particle positions orthogonal to the direction of the respectively used $\mathbf{q} \neq 0$ vector. Take for example $\mathbf{q} = q\mathbf{e}_x$. First $\rho(\mathbf{g}+\mathbf{q})$ is calculated through

$$\begin{aligned}
\rho(\mathbf{g}+\mathbf{q}) &= \sum_{\mu=1}^{N} e^{-i\mathbf{g}\cdot\mathbf{x}_\mu} e^{-i\mathbf{q}\cdot\mathbf{r}_\mu} \overset{\mathbf{q}=q\mathbf{e}_x}{=} \sum_{\mu=1}^{N} e^{-i\mathbf{g}\cdot\mathbf{x}_\mu} e^{-iqr_\mu^x} \\
&= \sum_{\{r^x\}} e^{-iqr_\mu^x} \sum_{i\in I_x} e^{-i\mathbf{g}\cdot\mathbf{x}_\mu} \approx \sum_{C_x} e^{-iqC_x} \sum_{i\in I_x^{\mathrm{bin}}} e^{-i\mathbf{g}\cdot\mathbf{x}_\mu} \\
&= \sum_{C_x} e^{-iqC_x} \rho_{\mathbf{g}}(C_x).
\end{aligned}$$

Here $\{r^x\}$ is the set of all x values in the system, I_x is the index set containing all indices of particles at position x, C_x is the center of the bins in real space and I_x^{bin} is the index set off all indices belonging to particles within the bin to the center C_x. The $\rho_{\mathbf{g}}(C_x)$ are obtained through a FFT. Now $\langle \rho^*(\mathbf{g}+\mathbf{q})\rho(\mathbf{g}'+\mathbf{q})\rangle$ can be calculated via multiplication and averaging over many snapshots. It becomes apparent that this method is a further approximation.
From this result one has to subtract $V^2 n_{\mathbf{g}} n^*_{\mathbf{g}'}$ for $\mathbf{q} = \mathbf{0}$ in order to get $\langle \delta\rho^*(\mathbf{g}+\mathbf{q})\delta\rho(\mathbf{g}'+\mathbf{q})\rangle$. Then the matrix is inverted. Unfortunately, the matrix is often singular for $q = 0$, which is why the $q = 0$ values will be omitted in the following.
Now the components of the $\mathbf{\Lambda}$-matrix can be calculated through equation (6.1). One specialty here is that, as mentioned before, the equilibrium of the system does not depend on T. It can therefore be chosen arbitrarily in this calculation and shall be factored out when discussing the results. Finally, to receive the dispersion relation, the eigenvalues of the $\mathbf{\Lambda}$-matrix are determined and plotted over the length of the wave vector q.

6.2 Dispersion Relation in Solid System

Choosing $\mathbf{q}$ as pointing from the Γ to the X point in a solid system yields the dispersion relation shown in Figure 6.1. The figure shows the square root of the eigenvectors normalized by the temperature and the Boltzmann-constant $\sqrt{\frac{\lambda}{k_B T}}$ plotted over q. Because the system is in the solid phase, it is ordered in both position and orientation. So the order in all three degrees of freedom leads to three different branches of eigenvalues in

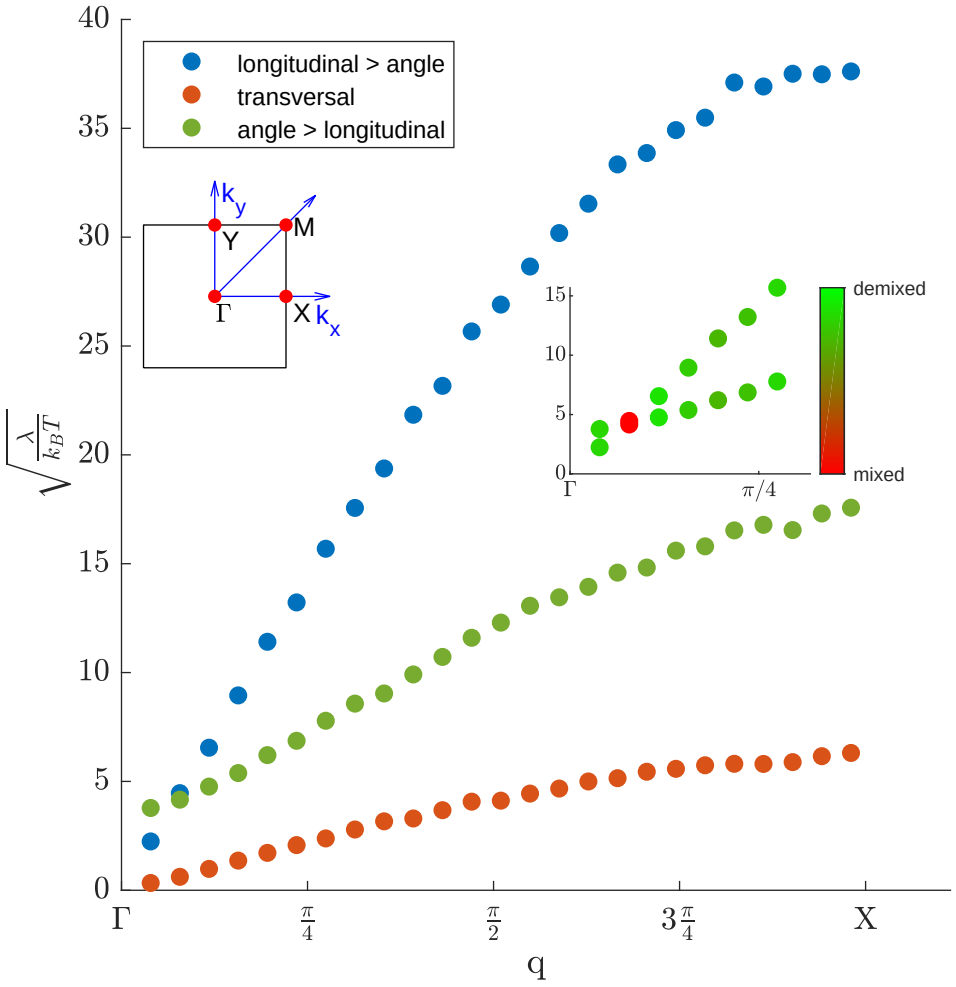

FIGURE 6.1: Dispersion relation with non vanishing angular branch. Depicted is the dispersion relation of a $N = 51^2$ with $\phi = 0.85$ and $\mathbf{q}$ going from the Γ to the X point. The first generation of $\mathbf{g}$ and 8000 decorrelated snapshots are used. λ are the eigenvalues of the $\mathbf{\Lambda}$-matrix. Three dispersion branches are visible, because the system is ordered in all three degrees of freedom. The transverse branch can clearly be identified by the eigenvector belonging to the eigenvalues, however the eigenvectors of the longitudinal and the angular branches mix. This mixing is shown in an inset and is given by the difference between absolute values of the eigenvector components in the longitudinal and angular direction. The classification of the longitudinal and angular branch is according to the dominating direction of the eigenvectors. The transversal and longitudinal branches tend to zero for $q \to 0$, the angular branch does not.

the dispersion relation. When the eigenvectors belonging to the respective eigenvalues are determined, the branches can be classified by the direction of the eigenvectors. The eigenvectors of the lowest branch always point in k_y direction and can therefore be identified as the transversal branch. For the other two branches the situation becomes less clear. Here, the eigenvectors do not always point in k_x or m direction, but sometimes they just lie in the k_x-m-plane. Luckily, they always tend to have a larger k_x or m component, so in Figure 6.1 they are classified by the dominating component as belonging to the longitudinal or angular branch. Figure 6.1 also contains an inset showing the mixing of the components of the longitudinal and the angular branch. This mixing is calculated as the absolute difference between the absolute values of the single components. Here, an interesting development shows itself. Note first that the longitudinal and the angular branches cross for small q.

For the smallest available q value, the two branches are demixed. But for the q value where the branches cross the components are heavily mixed. After the crossing the branches are again demixed.

One possible explanation for this behavior could be the following. As the two branches cross, the three dimensional matrix has only two eigenvalues. Therefore the algebraic multiplicity of the eigenvalues becomes two. Accordingly the geometric multiplicity and with it the dimensionality of the eigenspace can be one or two, allowing the eigenvectors to be defined on a plane.

Without being familiar with the algorithm used by matlab to compute eigenvalues it is suspected that it has problems determining eigenvectors to eigenvalues with an algebraic multiplicity larger than one.

Furthermore Figure 6.1 shows that the longitudinal and transversal branches scale linearly in q with $q \to 0$. Since the square root of the eigenvalues are plotted this implies that the respective entries of the $\mathbf{\Lambda}$-matrix scale with $\Lambda^{\alpha\beta} \propto \mathcal{O}(q^2)$. So therefore, as explained in Section 2.5.3, the theory yields the classical elasticity theory in the hydrodynamic limit for longitudinal and transversal oscillations. For the angular branch the scaling for $q \to 0$ can not clearly be identified.

Also very interesting in the $q \to 0$ behavior is whether the branches tend to zero or not. Since the value for $q = 0$ had to be disregarded, the behavior of the branches for $q \to 0$ can only be extrapolated from the existing data points.

The progression of the longitudinal and the transversal branches suggest that they indeed go to zero for $q \to 0$. A descriptive explanation for this is given in the following. If the wave vector goes to zero this means that the wave length goes to infinity. But a transversal or longitudinal wave with infinite wave length is simply a displacement of the whole system. This displacement does not invoke any restoring forces and hence no oscillation is induced. Therefore, the frequency goes to zero with wave vector to zero in this case.

A more uncommon behavior can be observed in the angular branch. It does not tend to zero with the wave vector going to zero. Using the argument from above, an angular $q \to 0$ wave consists of rotating all squares simultaneously. But in this case the squares will collide at some point in the rotation, creating a restoring force. The system has a quasi-long range positional order, meaning it has local crystalline structure in the particle positions. But this means that the collisions happen for very similar angles, creating a potential for the restoring force that is low for small angles and high for large ones. It is therefore possible to induce oscillations even for $q \to 0$, so the frequency does not go to zero.
The last feature of Figure 6.1 that shall be mentioned is the similar height of the longitudinal and the angular branch for small wave vectors q. At first glance one could think that this means that the frequencies of the angular and the longitudinal oscillations for these wave vectors are very similar. This deduction would be hard to explain, since a coupling of the transversal to the angular oscillation seems to be more likely because the shifting of lattice lines against each other could easily induce rotations of the squares. But fortunately, in this deduction the influence of the mass tensor in the wave equation (2.11) is not respected. The ratio of the moment of inertia and the mass of the squares is represented in this mass tensor. In a dispersion relation actually connecting frequency and wave vector it affects the ratio of the angular to the other two branches. So the deduction suggested above is not generally true.
It was planned to now discuss dispersion relations in different symmetry directions in the Brillouin-zone. Unfortunately, the fast method of calculating the $\mathbf{\Lambda}$-matrix via the algorithm provided by Johannes Häring does not yield sensible results.

6.3 Influence of the Reciprocal Lattice Vectors

Before, the dispersion relation is calculated using only the first generation of $\mathbf{g}$. Now the influence of using more $\mathbf{g}$ will be evaluated.
A dispersion relation in the solid phase using the first and second generation of $\mathbf{g}$ is shown in Figure 6.2.

This dispersion relation should be compared to the dispersion relation in Figure 6.1, as they are generated using the same data and only the number of $\mathbf{g}$ vectors is changed. Interestingly, both figures look very much alike. The only noticeable difference is the smoother behavior of the branches in Figure 6.2.
The fact that these two dispersion relations look so similar is not at all something that can be trivially expected a priori. After all, the restriction of the number of $\mathbf{g}$-vectors is an approximation. An exact solution of the wave equation would require all $\mathbf{g}$ to be taken into account. Therefore it can be inferred that a higher number of $\mathbf{g}$-vectors yields

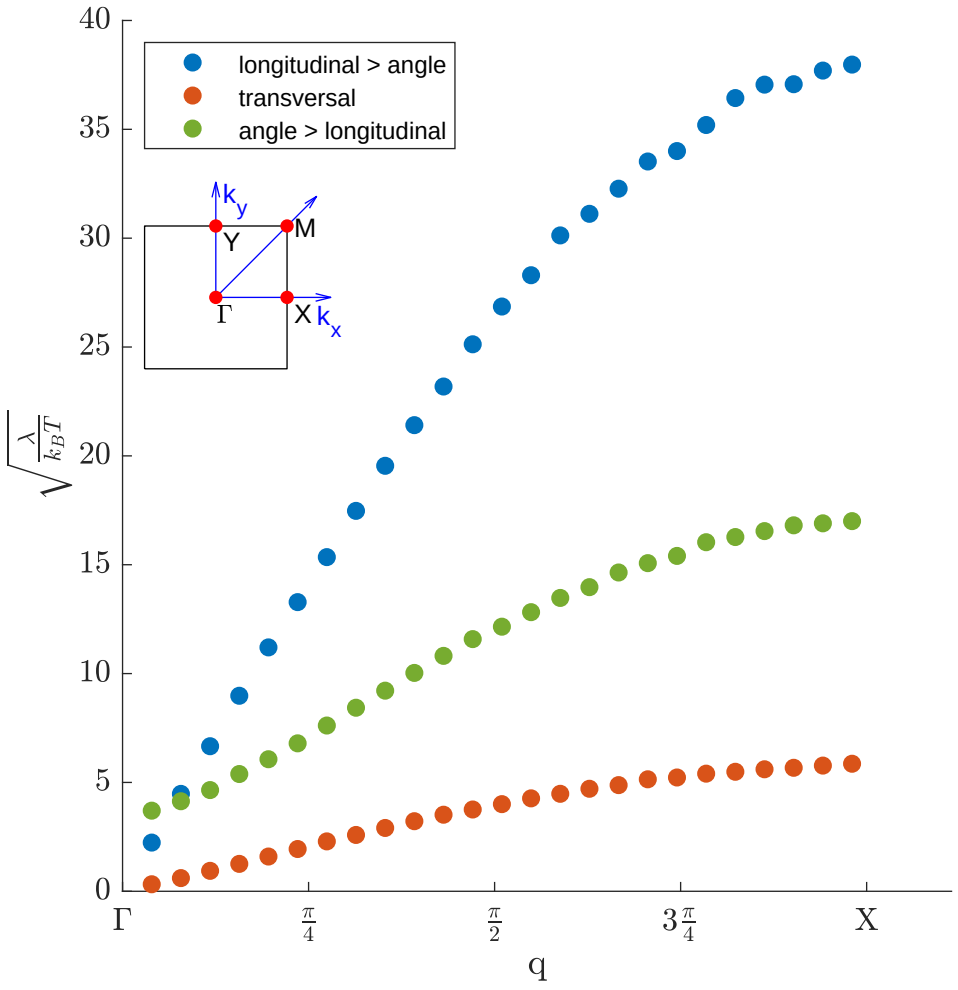

FIGURE 6.2: Influence of more **g** generations on the dispersion relation. Shown is a dispersion relation for a $N = 51^2$ and $\phi = 0.85$ system with **q** going from the Γ to the X point. The first and second generation of **g** and 8000 decorrelated snapshots are used. λ are the eigenvalues of the $\mathbf{\Lambda}$-matrix. The classification of the longitudinal, transversal and angular branch is according to the dominating direction of the eigenvectors. The data and parameters used for the creation of this plot are equal to the ones used in Figure 6.1. The only difference is the number of **g** vectors.

a better approximation.

In the present case however, the improvement of the approximation only yields smoother data but does not change behavior of the dispersion branches or their overall amplitude. Therefore only using the first generation of **g**-vectors is already an adequate approximation, if one is only interested in the behavior and not the smoothness of the curves. Since it is computationally far less expensive to only choose the first generation of **g**-vectors and since the above argues that this poses no restriction to the analysis of the dispersion relation, the following dispersion relations shall be computed using only the first generation of **g**-vectors.

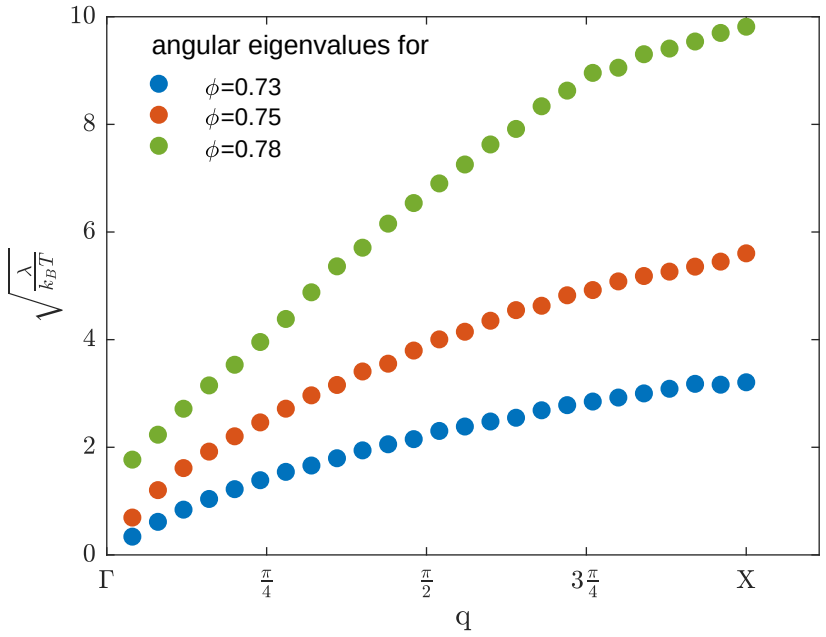

FIGURE 6.3: Behavior of the angular branch in different phases of the system. Plotted is the dispersion relation of the angular branch for different packing fractions in a $N = 51^2$ system from the Γ to the X point. The first generation of $\mathbf{g}$ and 8000 decorrelated snapshots are used. The eigenvalues for $\phi = 0.73$ and $\phi = 0.75$ are calculated with the method described in Section 6.4. The eigenvalues for $\phi = 0.78$ are calculated with the method described in 6.1. Note that the system is in the tetratic phase for $\phi = 0.73$ and $\phi = 0.75$ and in the solid phase for $\phi = 0.78$. For $\phi = 0.73$ and $\phi = 0.75$ the eigenvalues tend to zero with $q \to 0$. For $\phi = 0.78$, they do not.

6.4 Dispersion Relation in Tetratic System

The dispersion relation can also be obtained in the tetratic phase of the system. This is conceptually done in the same way as in the previous section. But one has to be mindful of the fact that there is only short range positional order in the tetratic phase. Accordingly there exist no Bragg-peaks and no reciprocal lattice structure. The orientational order is still quasi-long range, so the only defined components of the $\mathbf{g}$ vectors are the coefficients of the Fourier series m. Looking at definition (6.1) this shows that only the angular Λ^{33} entry of the $\mathbf{\Lambda}$-matrix is defined. So the dispersion relation in the tetratic phase will only exhibit one branch. Two of these branches for different packing fractions in the tetratic phase and one angular branch from the solid phase are displayed in Figure 6.3.

Here, the two branches in the tetratic phase $\phi = 0.73$ and $\phi = 0.75$ are calculated by the method that is described in this section. For $\phi = 0.78$, the system is already solid. The eigenvalues generally increase for higher packing fractions, indicating higher frequencies and steeper scalings. This can be interpreted as the system behaving 'softer' for lower packing fractions and 'harder' for higher ones.

A very interesting property of Figure 6.3 is the following. It shows the angular branch in the $\phi = 0.78$ system not going to zero in the hydrodynamical limit, as is also the case in Figure 6.1. But as the orientational order goes from being long range in the solid phase to quasi-long range in the tetratic phase, the eigenvalues for the $\phi = 0.73$ and $\phi = 0.75$ start to go to zero for $q \to 0$. Using similar arguments as for the behavior in the solid phase, this can also be explained. Again, a $q \to 0$ rotation rotates all particles at once. Indeed this will still create collisions, but since the positional order in the system is lost, the collisions will happen for arbitrary angles. This does not allow for restoring forces that induce oscillation and hence the frequency goes to zero.
The vanishing of the $q \to 0$ value of the angular branch can be connected to the symmetry of the system, ergo to the phases. This vanishing behavior indicates that there exist low frequency, so low energy, high wave length angular modes in the system. But because these modes can be easily excited they should destroy any angular long range order in the system. And indeed this behavior is only observable in the tetratic phase, where long range orientational order is lost. This is very similar to what is observed in the Mermin-Wagner fluctuations destroying any long range positional order.

6.5 Finite Size Analysis

As usual the dispersion relation should also be subjected to a finite size analysis. Therefore the dispersion relation of a $N = 21^2$, $\phi = 0.85$ system is plotted in Figure 6.4. Generally the behavior of the branches looks very similar to Figure 6.1. The only noticeable difference are the slightly varying overall amplitudes of the dispersion branches. These variations are probably caused by the angular and positional symmetry being affected by the finite size of the system.
Through the similarity between Figures 6.1 and 6.4 one can infer that the particular elastic properties of the system of hard squares can already be found in small sample sizes.
Note that the system size restricts the number of possible q-vectors and therefore the number of depicted data points in Figure 6.4.

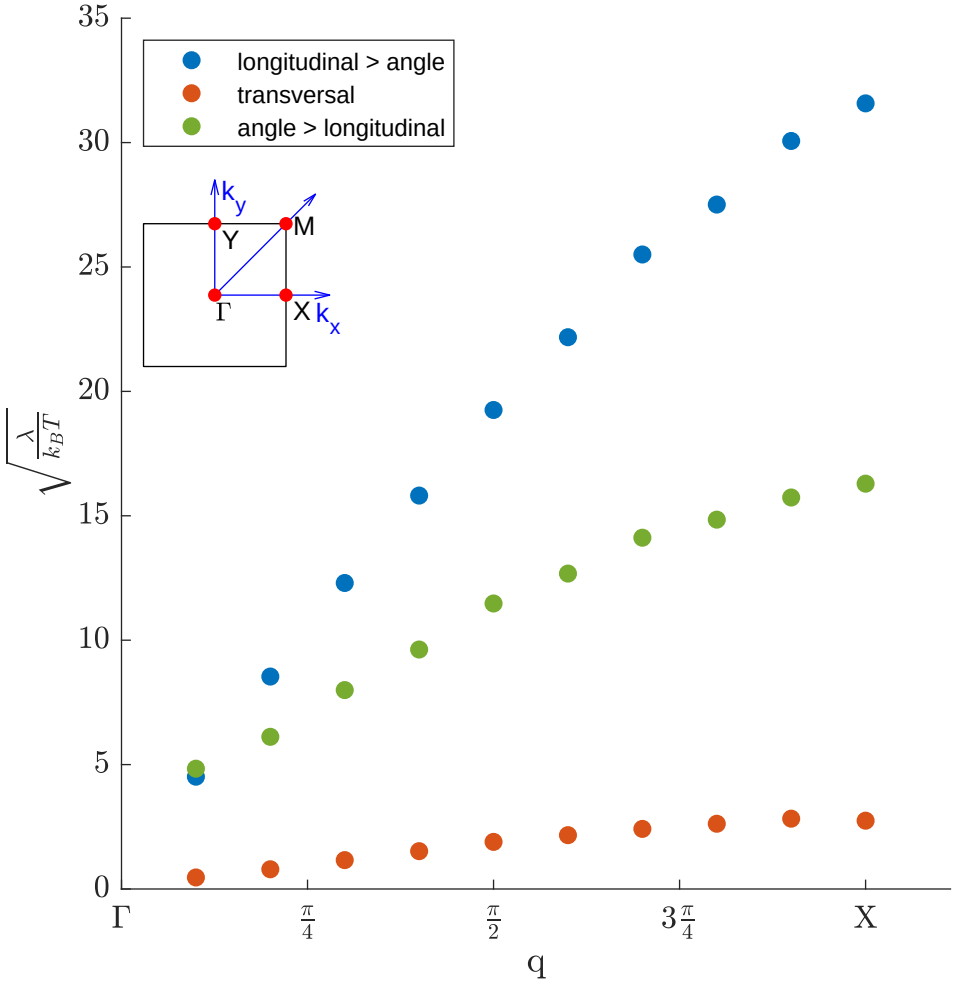

FIGURE 6.4: Finite size scaling of the dispersion relation. Depicted is a dispersion relation for a $N = 21^2$ and $\phi = 0.85$ system with $\mathbf{q}$ going from the Γ to the X point. The first generation of $\mathbf{g}$ and 3000 decorrelated snapshots are used. This figure is produced to allow a finite size analysis in comparison to Figure 6.1. λ are the eigenvalues of the $\mathbf{\Lambda}$-matrix. The classification of the longitudinal, transversal and angular branch is according to the dominating direction of the eigenvectors.

Chapter 7

Summary and Outlook

7.1 Summary

In this thesis, Monte-Carlo simulations of two-dimensional hard squares systems were successfully implemented. The data produced yielded consistent results within the analysis methods. The simulation algorithm was successfully optimized by the implementation of cell lists, which increased the computation speed by a factor of up to 30.
The existence of the tetratic phase in the system of hard squares, which has already been found by Glotzer *et al.* [1] and Wojciechowski and Frenkel [3], was confirmed. A computation of the phase transitions via the sub-block analysis yielded results well in accordance with the findings of Glotzer *et al.* [1]. These phase transitions were verified by consideration of the scaling of the pair correlation functions defined for the local order parameters. For polydisperse systems a stability of the tetratic phase was found, showing that it exists in systems with polydispersities of up to $pd = 17\%$.
An investigation of point defects in the reciprocal space produced a vacancy density. This vacancy density rises to concentrations of up to 1.4% with decreasing packing fractions. This concentration is considerably higher compared to what is found by Zollweg *et al.* in the two dimensional systems of hard disks [34]. An algorithm to identify the local lattice structure was implemented using an enhanced Voronoi construction. Due to time constraints the resulting data was not analyzed properly. It is hoped that topological defects could be recognized using this data and the Burgers-vector construction.
With the above, the preliminary work was completed. The main goal of calculating the dispersion relations of the lattice vibrations via an elasticity theory for real crystals [2] was achieved. It was shown that the angular branches of the dispersion relation do not tend to zero for $\mathbf{q} \to 0$ in the solid phase. However, in the tetratic phase they do. This can be linked to the long range orientational order in the solid phase and the quasi-long

range orientational order in the tetratic phase via the existence of low energy and long range modes.

7.2 Outlook

There is a lot of work that could still be done in the thematic vicinity of this thesis. Some of it is presented here.
Regarding point defects, the periodic boundary conditions could be improved to have a variable form. This has been done in other work [3] and could be implemented quite quickly. It would help produce point defect densities that have values close to what one might expect in the bulk of infinitely large systems while also making the finite size analysis more fruitful. Furthermore, the data produced by the Burgers-vector construction using the local lattice structure could be evaluated in more detail. One could aim to receive a density of topological defects. The elasticity theory for real crystals can also produce the second moment of the defect density [36]. This could be compared to the data obtained in this thesis or by future work.
For the dispersion relations one improvement is apparent. The dispersion relations in different symmetry directions should be calculated in order to complete the analysis. It would also be interesting to find the underlying symmetry argument explaining the differences in the behavior of the angular branch in the hydrodynamic limit in the solid and the tetratic phase.
Looking at future experiments, the developments in the SFB1214 are very interesting. There are planned experiments of colloidal cubes, made of neighborite or silver, sedimented into monolayers [5]. These systems are very similar to the system discussed in this thesis. The analysis code developed in this thesis could be applied to data from these experiments. Comparing the results promises to yield interesting insights, especially concerning the behavior for large numbers of particles.

Appendix A

Preparation of the Snapshots for Averaging

Sometimes, one needs to average over the equilibrium particle positions in the system. But in order for the averages to yield meaningful results, any drift or rotation that the whole system performs must be counteracted. This is especially important when calculating the dispersion relation, so that the Bragg-peaks do not smear out. The basic idea in the following approach is to use the properties of the Fourier transformation under translation to find the displacement of the real lattice from an ideal, centered lattice.

Consider a real system of squares with a drifting center of mass. Let the position of the center of mass be $\mathbf{c}$. Then the vector r can be rewritten as $\mathbf{r} = \tilde{\mathbf{r}} + \mathbf{c}$. If one now calculates the Fourier transform of the density, one finds using the behavior of the Fourier transformation under translation in real space

$$\mathcal{F}(\rho(\tilde{\mathbf{r}} + c))(k) = e^{i\mathbf{c}\cdot\mathbf{k}}\mathcal{F}(\rho(\tilde{\mathbf{r}}))(k),$$

where $\rho(\tilde{\mathbf{r}})$ is the density of the system with the center of mass in the origin. Therefore, the displacement of the center of mass can be calculated in components by

$$c = \frac{1}{ik}\ln\left(\frac{\mathcal{F}(\rho(\tilde{\mathbf{r}} + c))(k)}{\mathcal{F}(\rho(\tilde{\mathbf{r}}))(k)}\right) = \frac{1}{ik}\ln\left(\frac{\mathcal{F}(\rho(\mathbf{r}))(k)}{\mathcal{F}(\rho(\tilde{\mathbf{r}}))(k)}\right). \tag{A.1}$$

So if one knows the Fourier transform of the density of the non-displaced lattice, one can calculate the displacement. In the following it will be described how this information is obtained, and how the lattice is centered for averaging.

As mentioned above, one needs to find the Fourier transform of the non-displaced lattice. In order to obtain this, the following steps are taken. First, the Fourier transform of

the real system regarding discrete wave vectors is taken. One now needs to extract the position of two orthogonal Bragg-peaks from the reciprocal image. This is done by

1. Identify the constant peak and measure its height. Then determine the area around the constant peak that has larger values than γ the height of the constant peak. Set this area to zero.
2. Compute the global maximum of the image. Save position $\mathbf{k}_1$ and height h_1 of that maximum. Then determine the area around $\mathbf{k}_1$ that has larger values than $\gamma \cdot h_1$. Set this area to zero.
3. Compute the global maximum of the image. Save position $\mathbf{k}_2$ and height h_2 of that maximum. Then determine the area around $\mathbf{k}_2$ that has larger values than $\gamma \cdot h_2$. Set this area to zero. If $|\mathbf{k}_1 \cdot \mathbf{k}_2| > \delta$, go back to 3.
4. $\mathbf{k}_1$ and $\mathbf{k}_2$ are the positions of two orthogonal Bragg-peaks.

The position of these two orthogonal Bragg-peaks give the two reciprocal lattice vectors $\mathbf{b}_i = \mathbf{k}_i$. The lattice vectors a_i in real space can be computed by solving [22]

$$\mathbf{A} = 2\pi \left(\mathbf{B}^{\mathrm{T}}\right)^{-1}, \quad \text{with} \quad \mathbf{A} = \begin{pmatrix} a_1^x & a_2^x \\ a_1^y & a_2^y \end{pmatrix},$$

and $\mathbf{B}$ accordingly. With these real space lattice vectors one can firstly calculate the net rotation of the real system. Ideally, the system should then be rotated back whilst fulfilling the periodic boundary conditions. Unfortunately this is not possible if the rotation is larger that an few minutes. Visualizing the periodic boundary conditions as the system sitting on a three-dimensional torus, it becomes clear that artificial rotations of the whole system are not possible without violating the periodic boundary. It should be mentioned that this does not mean that the system can not rotate in the Monte-Carlo simulation. Here, the system can rotate bit by bit, whilst restructuring in between to respect the boundary. This behavior is also observed in Section 3.6.
As a solution to this problem it was chosen to rotate the system neglecting the periodic boundary conditions, but to discard any simulation runs where the system visibly rotated out of the reference of the boundary box sides and stayed there for a prolonged amount of simulation time.
If a rotation of the system is necessary, the Fourier transformation of the real system is also rotated by that amount.
Now the real space lattice vectors are used to construct a perfect and centered lattice in

real space. The Fourier transform of this lattice is then taken to represent the Fourier transform of the non-displaced lattice $\mathcal{F}(\rho(\tilde{\mathbf{r}}))(k)$.
Since now the Fourier transform of the real lattice and the non-displaced lattice are known, equation (A.1) can be used to calculate the components of the displacement c. Even though the equation should hold true for all wave vectors, an evaluation at the Bragg-peak positions $\mathbf{k}_1$ and $\mathbf{k}_2$ found earlier generates the best results.
Lastly, the periodic boundary conditions are applied after displacing the system by c.

Appendix B

Mathematical Methods

B.1 Fourier Transform

A great portion of the analysis in this thesis is done in reciprocal space. It can be accessed via the Fourier transform. The convention used in this thesis for the Fourier transform of some function $f(\mathbf{r})$ of real space and the reciprocal wave vector $\mathbf{k}$ is

$$\mathcal{F}(f(\mathbf{r}))(\mathbf{k}) = \int_{\mathbb{R}^2} \mathrm{d}\mathbf{r} \quad e^{-i\mathbf{k}\cdot\mathbf{r}} f(\mathbf{r}).$$

The inverse Fourier transform of some function $\tilde{f}(\mathbf{k})$ of the reciprocal space is given accordingly by

$$\mathcal{F}^{-1}(\tilde{f}(\mathbf{k}))(\mathbf{r}) = \frac{1}{(2\pi)^2} \int_{\mathbb{R}^2} \mathrm{d}\mathbf{k} \quad e^{i\mathbf{k}\cdot\mathbf{r}} \tilde{f}(\mathbf{k}).$$

B.2 Spatial Tessellations

The Voronoi construction and the Delaunay triangulation are used throughout the thesis, mainly for the acquisition of next neighbors. The definition of these spatial tessellations is given here.

B.2.1 Voronoi Construction

The Voronoi construction has many applications in varying fields. This versatility comes from the simplicity of the construction and the mostly apparent interpretation. In the following the Voronoi construction will be formally defined in two dimensions and then shortly explained.

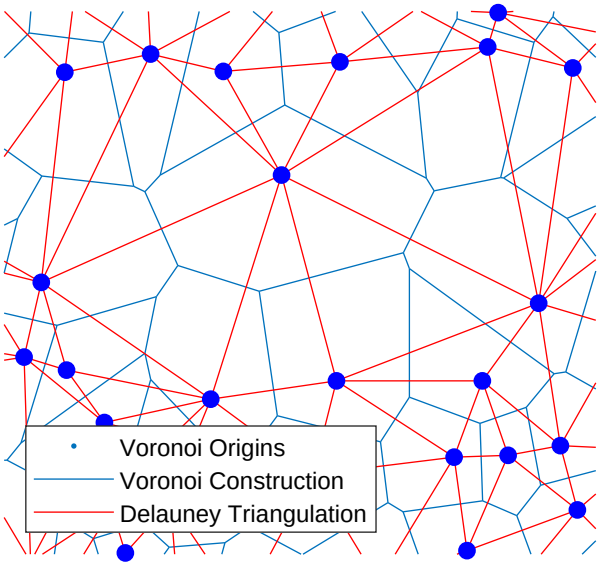

FIGURE B.1: A Voronoi construction (blue) and its dual, the Delaunay triangulation (red) of random points. Note how the faces of the Voronoi cells mark all points in space where the distance to two Voronoi origins $\mathbf{p}_\mu$ is equally small. On the vertices, the distance to three such points is equal. A Delaunay bond exists between two $\mathbf{p}_\mu$ sharing a face in the Voronoi construction.

Let $P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\} \subset \mathbb{R}^2$, $2 < n < \infty$, $\mathbf{p}_\mu \neq \mathbf{p}_\nu$ for all $\mu \neq \nu$ with $\mu, \nu \in I_n$. Then a Voronoi cell of Voronoi polygon is defined by [37]

$$V(\mathbf{p}_\mu) = \left\{\mathbf{x} \in \mathbb{R}^2 \,|\, \forall \nu \in I_n,\, \nu \neq \mu : \, ||\mathbf{x} - \mathbf{p}_\mu|\,|_2 \leq ||\mathbf{x} - \mathbf{p}_\nu|\,|_2\right\}.$$

The Voronoi construction is defined as

$$\mathcal{V} = \{V(\mathbf{p}_\mu)\}.$$

This construction has a very simple purpose. It takes a finite set of points in two dimensions P and divides the two-dimensional space into Voronoi cells $V(\mathbf{p}_\mu)$. In each cell the points are closer to the center of the cell than to any other cell centers in respect to the euclidean norm (see Figure B.1).

The most important feature for this thesis is that if two Voronoi cells share a face, their origins $\mathbf{p}_\mu$ and $\mathbf{p}_\nu$ can be seen as next neighbors. This means there exists a set of points, namely those lying on the shared face, which are equally close to both origins $\mathbf{p}_\mu$ and $\mathbf{p}_\nu$. This next neighbor connection between particle positions will be called a bond. Since these bonds can be obtained for each particle position, they themselves form a spatial tessellation.

B.2.2 Delaunay Construction

The Delaunay construction consists of the bonds between origins $\mathbf{p}_\mu$ that can be found with the Voronoi construction. It is therefore often called dual to the Voronoi construction. This duality means that for each Delaunay bond there exists a unique face in the Voronoi construction. Generally, the Delaunay construction divides the two dimensional space into triangles, which is why it is also called a triangulation. This triangulation can be very helpful in finding the local structure of the system, since it visualizes all the bonds in the system (see Figure B.1). [37]

Appendix C

On Symmetries and Order Parameters

In Section 2.1.2 the definition of order parameters is discussed. Even though it is in principle correct, the given definition for order parameters omits their connection to the symmetry in the system. This connection is not really relevant for the evaluation of results in this thesis, but nevertheless it is important to discuss it in order to understand the definition in a greater context.

Again: the Ising Model The following discussion will directly connect to the discussion of the Ising model in Section 2.1.2.
The Hamilton-operator of the Ising model in an external magnetic field is given by

$$\mathcal{H} = -\frac{1}{2}\sum_{l'l} J(l-l')\sigma_l\sigma_{l'} - h\sum_l \sigma_l. \tag{C.1}$$

The indices $\{l'l\}$ denote all lattice positions and $J(l-l')$ is the exchange interaction only dependent on the distances between the lattice positions l and l'. $\sigma_l = \pm 1$ are the eigenvalues of the Pauli-matrices and denote whether a spin is positioned parallel or antiparallel to the z-axis and h contains the magnetic field strength H of the external magnetic field. The underlying symmetry of the system can be analyzed by looking at the Hamilton-operator (C.1). For example, if $h = 0$ then transforming $\sigma_l \to -\sigma_l$ will induce $\mathcal{H} \to \mathcal{H}$. This means that the Hamilton-operator is invariant under an inversion in the z-axis. For any system with a Hamilton-operator these transformations under which the Hamilton-operator is invariant constitute a symmetry group $\mathcal{G}$ of the Hamilton-operator.
If one now regards the symmetries of configurations of the system it is clear that they

must be contained in the symmetry group $\mathcal{G}$ of the Hamilton-operator. This means that actual configurations always have fewer or the same symmetries as the Hamilton-operator. But how is this connected to the order parameters?

Suppose one looks at a configuration of a system. This configuration has a certain amount of symmetry operations it is invariant under, and these operations are contained in $\tilde{\mathcal{G}}$. As a concrete example take the Ising model system in the paramagnetic phase. Since the spins are oriented without a preferred direction, the configurations of the system in the paramagnetic phase are symmetric in respect to inversion regarding the z-axis. This inversion would be contained in this configuration's $\tilde{\mathcal{G}}$.

Generally speaking, if a configuration of a system has a certain symmetry, then all non-zero averages must be invariant under this symmetry. This can be made plausible by stating that it would not be reasonable to have a system that is microscopically invariant under some transformation, but the same transformation changes a macroscopic variable of the system. Again, look at the Ising model. As stated, the paramagnetic phase is inversion symmetric in z. The magnetization $m = \langle \sigma_l \rangle = 1/N \sum_l \sigma_l$ is generally not invariant under the same transformation, because $\sigma_l \to -\sigma_l$ implies $m \to -m$. Therefore, one can already state that m must be equal to zero in the paramagnetic phase.

Now let the system perform a phase transition. In the Ising model this would mean cooling to a temperature below T_C to enter the ferromagnetic phase. As a phase transition always entails a change in internal symmetry, the transformations contained in the symmetry group of the new phase $\mathcal{G}'$ are not the same as in $\tilde{\mathcal{G}}$. Suppose the phase transition induced order, then some transformations must have vanished from $\tilde{\mathcal{G}}$ to form the symmetry operations of the new phase $\mathcal{G}'$. Referring to the Ising model, the inversion symmetry by z is lost in the ferromagnetic phase. This is because at the phase transition the system underwent a spontaneous breaking of symmetry and the spins now have a random but fixed preferred direction. Since inversion in z would change the sign of this fixed direction, the configuration is not inversion symmetric in z.

Generally speaking this loss of symmetry operations enables macroscopic variables that where forced into being zero by the higher symmetry of the previous phase to become non zero. Exactly these variables are the order parameters. Again, for the Ising model this means that the loss of inversion symmetry in z in the ferromagnetic phase allows for $m \neq 0$[1]. Therefore m is the order parameter of the ferromagnetic phase. The behavior predicted here for m is exactly what is found in equation (2.1).

In this way the order parameters of a system are deeply connected to the symmetries in the different phases of the system. An understanding of this connection can facilitate a better understanding of phases themselves.

[1] Of course all the other symmetries [38] of both phases have to be checked. Anyhow for m this is still true.

Equipped with the preceding explanation one can now tackle the definition of order parameters as found in textbooks.

Alternative Definition of Order Parameters Suppose a system is in a configuration of high symmetry. Any thermodynamical averages that are non-zero must be invariant under the transformations contained in the symmetry group $\tilde{\mathcal{G}}$ of this system configuration. If the system now enters a state of lower symmetry, this state of higher order allows the appearance of new non-zero averages $\langle O \rangle$. These averages $\langle O \rangle$ are not invariant under $\tilde{\mathcal{G}}$ and therefore must also be zero in the state of high symmetry. $\langle O \rangle$ are called order parameters. But how do order parameters and phase transitions relate? The order parameter is non-zero if the system is in a state of higher order and becomes zero when the order of the system is reduced. Therefore a phase transition occurs when an order parameter tends to zero. [21]

Appendix D

Optimization of the Matlab Code

Matlab (Mat*rix* Lab*oratory*) is chosen as the programming language in this thesis. The benefits of matlab are easy and forgiving usage, fast and good representation of data and quick prototyping of code. Some drawbacks are the sometimes bad or missing documentation and the missing ability to access matlab core functions.
When using matlab, one has to be mindful of another potential drawback. The internal structure of matlab as an interpreter language [39] can lead to major speed decreases when loops over many iterations are executed. Indeed, in this thesis this often leads to problems as, for example, the Monte-Carlo routine can loop over tens of billions of iterations. Two solutions to this problem where implemented in this thesis.

Transition to C-Code Matlab has a build in feature that allows for the automatic creation of C-code. So the code is written as usual in matlab, using the problematic long loops, and then translated by matlab into C-code. Since C is a compiler language [40], it does not exhibit the same problems for large loops. The compiled C-code can be embedded into a framework of matlab code. In the case of the Monte-Carlo routine, the use of C compiled code leads to speed increases of up to two orders of magnitude.
The method generally gives very good results and is easy to use the only problem being that already compiled C-code can not be debugged in matlab.
Also, the matlab C-coder does not support parallelization, so it can only run on a single core. For the Monte-Carlo routine, this is not problematic, since it is implemented without any parallelization methods. But when considering the evaluations needed for the dispersion relation, where many nested loops perform the same basic calculation, parallelization is the perfect choice for increasing computation speeds.

Parallelization Matlab offers the so called '*parfor*'-loop, which can outsource the calculation of loops over many similar but independent iterations to a number of processing core. This is generally only useful for very large computations, as the distribution of the loop calculation to different cores itself takes a lot of computation time. It is found that for the dispersion relation evaluation this did not create any major speed increases, but on the contrary sometimes even slowed the computations down. But since matlab was conceived for calculating with matrices, hence the name, it offers another solution closely related to parallelization.

Vectorization Most loops formulated can be replaced by using an intricate combination of vectors, matrices and vector products. Matlab's internal workings can translate this use of matrices into a highly efficient, parallelized code. This is called vectorization. In the following an example for such a vectorization is given.
Consider the calculation of the Bragg-peak order parameters before averaging $\tilde{n}_{\mathbf{g}}$. So one wants to calculate

$$\tilde{n}_{\mathbf{g}} = \sum_{i=1}^{N} e^{-i\mathbf{g}\cdot\mathbf{x}_i}$$

for a single $\mathbf{g}$ and for particle positions and angles $\mathbf{x}_i$. Using for-loops, the matlab code for this calculation would be

```
for i=1:N
    n=n+exp(-1i*(g'*x(:,i)));
end
```

Here `g'*x(:,i)` is the scalar product $\mathbf{g}\cdot\mathbf{x}_i$.
This code can now be vectorized by using the whole matrix `x(:,:)` in the scalar product. The vectorized code is

```
i=1:N;
n=sum(exp(-1i*(g'*x(:,i))));
```

This vectorization works by letting `exp(-1i*(g'*x(:,i)))` produce a row vector of all summands and then summing with the vector supportive `sum()` function.
This is an easy example for the use of vectorization, but it can become very complicated to produce vectorized code.
Matlab will automatically run vectorized code on many cores, from which the speed increase of vectorization originates.
The vectorization of large loops reliably increases the computation speed of code.

Specialties in Matlab In this paragraph some pitfalls in matlab that where found during the creation of this thesis shall be explained, so that the reader does not repeat them in order to find them.
The basic matlab memory allocation structure is based on fortran, which is why in loops, the indexing should always look like the following, even for preallocated arrays.

```
j=1:M
    i=1:N
        A(i,j)=c;
    end
end
```

and never like this

```
i=1:N
    j=1:M
        A(i,j)=c;
    end
end %false!
```

In the second variant, matlab has to skip chunks of memory in each iteration, making this implementation computationally more costly.
It is important to always initialize the random number generator. The matlab documentation suggests to do so by calling `rng('shuffle')`, if no special seed is required. However, the fine print of the documentation states that `rng('shuffle')` sets the system time as the seed. So if two simulations are initialized by calling this function in the same system second, they will have identical random number generator seeds, and will therefore produce exactly the same results. It is therefore good practice to always use a predefined set of different seeds to initialize the random number generator.
The quadrant dependent arcustangens function is defined as `atan2(Y,X)` with the Y coordinate as the first argument.

Appendix E

Algorithms

E.1 Orientational Trial Move

Algorithm 1 Compute Rotational Trial Move

function NEWORIENT(φ) ▷ φ is the angle before the trial move

$\beta \leftarrow \text{rand}(0,1) \cdot 2\pi$

$r_v \leftarrow \begin{bmatrix}\cos(\beta)\\ \sin(\beta)\end{bmatrix}$ ▷ create random point on unit circle

$d \leftarrow \Delta_\alpha r_v + \begin{bmatrix}\cos(\varphi)\\ \sin(\varphi)\end{bmatrix}$ ▷ create new vector for direction

$d \leftarrow \frac{d}{\text{norm}(d)}$ ▷ normalize d

$\varphi_n \leftarrow \arctan(d)$ ▷ compute angle after trial move

Return φ_n ▷ [27]

end function

E.2 Monte-Carlo Algorithm

Algorithm 2 Monte-Carlo Algorithm

```
procedure MC(sqs)                          ▷ sqs contains definitions for all squares
    linkl ← newList(sqs)                                          ▷ create cell list
    for i ← 1 : L do                       ▷ L is the number of Monte-Carlo steps
        z = randi(1 : N)                                  ▷ choose one square at random
        d = randi(0 : 1)                                  ▷ rotation or displacement move
        if d = 0 then
            nsq ← displace(sqs(z))                                ▷ displace chosen square
            nsq ← periodic(nsq)                       ▷ apply periodic boundary condition
            if enList(nsq,sqs,linkl)= 0 then                  ▷ enList = 0 if no overlaps
                linkl = upList(linkl,nsq,sqs)                         ▷ update cell list
                sqs(z) ← nsq                  ▷ if move accepted, replace chosen square
            end if
        else
            nsq ← 0 [sqs(z),NewOrient(sqs(z))]                    ▷ rotate chosen square
            if enList(nsq,sqs,linkl)= 0 then
                linkl = upList(linkl,nsq,sqs)
                sqs(z) ← nsq
            end if
        end if
    end for
end procedure
```

Acknowledgements

First I want to thank Prof. Fuchs and Johannes Häring for suggesting the topic of this thesis and their guidance through the same.
Similarly, my thanks go to the entire LS Fuchs for always helpfully answering my questions and participating in discussions.
I also want to thank Robert Löffler for showing me the challenges of the defect detection in this system. I personally enjoyed the tinkering on microscopic solutions very much.
I thank my family for giving me the chance to study physics by both their moral and financial support.
Many thanks to Luzia Marek, Corinna Dannert, David Ohnmacht and Markus Gruber for proof reading.

List of Figures

List of Tables

Bibliography

[1] Joshua A. Anderson, James Antonaglia, Jaime A. Millan, Michael Engel, and Sharon C. Glotzer. Shape and symmetry determine two-dimensional melting transitions of hard regular polygons. *Physical Review X*, 7(2), May 2017. doi: 10.1103/physrevx.7.021001.

[2] C. Walz and M. Fuchs. Displacement field and elastic constants in nonideal crystals. *Physical Review B*, 81(13), 2010. doi: 10.1103/physrevb.81.134110.

[3] K.w. Wojciechowski and D. Frenkel. Tetratic phase in the planar hard square system? *Computational Methods in Science and Technology*, 10(2):235–255, 2004. doi: 10.12921/cmst.2004.10.02.235-255.

[4] Lee Walsh and Narayanan Menon. Ordering and dynamics of vibrated hard squares. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(8):083302, 2016. doi: 10.1088/1742-5468/2016/08/083302.

[5] Robert Clemens Löffler. Phase behavior of 2d monolayers of cubic colloids. Master's thesis, Universität Konstanz, 2018.

[6] Werner Krauth. *Statistical mechanics: algorithms and computations*. Oxford University Press, 2012.

[7] N. D. Mermin. Crystalline order in two dimensions. *Physical Review*, 176(1): 250–254, May 1968. doi: 10.1103/physrev.176.250.

[8] Ken Bagchi, Hans C. Andersen, and William Swope. Computer simulation study of the melting transition in two dimensions. *Physical Review Letters*, 76(2):255–258, Jan 1996. doi: 10.1103/physrevlett.76.255.

[9] S. T. Chui. Grain-boundary theory of melting in two dimensions. *Physical Review Letters*, 48(14):933–935, May 1982. doi: 10.1103/physrevlett.48.933.

[10] Yves Lansac, Matthew A. Glaser, and Noel A. Clark. Discrete elastic model for two-dimensional melting. *Physical Review E*, 73(4), Mar 2006. doi: 10.1103/physreve.73.041501.

[11] J M Kosterlitz and D J Thouless. Ordering, metastability and phase transitions in two-dimensional systems. *Journal of Physics C: Solid State Physics*, 6 (7):1181–1203, Dec 1973. doi: 10.1088/0022-3719/6/7/010.

[12] The nobel prize in physics 2016. URL https://www.nobelprize.org/nobel_prizes/physics/laureates/2016/. Accessed: 2018-08-15.

[13] Hans-Hennig Von Grünberg, Peter Keim, and Georg Maret. Phase transitions in two-dimensional colloidal systems. *Soft Matter, Volume 3*, page 41–86, Jul 2014. doi: 10.1002/9783527682300.ch2.

[14] Michael Engel, Joshua A. Anderson, Sharon C. Glotzer, Masaharu Isobe, Etienne P. Bernard, and Werner Krauth. Hard-disk equation of state: First-order liquid-hexatic transition in two dimensions with three simulation methods. *Physical Review E*, 87(4), Apr 2013. doi: 10.1103/physreve.87.042134.

[15] Sander Pronk and Daan Frenkel. Point defects in hard-sphere crystals†. *The Journal of Physical Chemistry B*, 105(28):6722–6727, May 2001. doi: 10.1021/jp010779e.

[16] Etienne P. Bernard and Werner Krauth. Two-step melting in two dimensions: First-order liquid-hexatic transition. *Physical Review Letters*, 107(15), Jul 2011. doi: 10.1103/physrevlett.107.155704.

[17] F. Smallenburg, L. Filion, M. Marechal, and M. Dijkstra. Vacancy-stabilized crystalline order in hard cubes. *Proceedings of the National Academy of Sciences*, 109 (44):17886–17890, Oct 2012. doi: 10.1073/pnas.1211784109.

[18] Clement John Adkins. *Equilibrium thermodynamics*. McGraw-Hill, 1975.

[19] Franz Schwabl. *Statistische Mechanik*. Springer, 2006.

[20] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Physical Review*, 65(3-4):117–149, Jan 1944. doi: 10.1103/physrev.65.117.

[21] P. M. Chaikin and T. C. Lubensky. *Principles of condensed matter physics*. Cambridge University Press, 2010.

[22] Rudolf Gross and Achim Marx. *Festkorperphysik*. De Gruyter, 2014.

[23] N. D. Mermin and H. Wagner. Absence of ferromagnetism or antiferromagnetism in one- or two-dimensional isotropic heisenberg models. *Physical Review Letters*, 17 (26):1307–1307, 1966. doi: 10.1103/physrevlett.17.1307.

[24] Neil W. Ashcroft and David N. Mermin. *Solid state physics*. Saunders College, 1976.

[25] Monte carlo method, Jun 2018. URL https://en.wikipedia.org/w/index.php?title=Monte_Carlo_method&oldid=846165133. Accessed: 2018-08-10.

[26] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97, 1970. doi: 10.2307/2334940.

[27] Daan Frenkel and Berend Smit. *Understanding molecular simulation: from algorithms to applications*. Academic Press, 2012.

[28] Michael Tanaya, Hua Ming Chen, Jebediah Pavleas, and Kelvin Sung. *Building a 2D game physics engine: using HTML5 and JavaScript*. Apress, 2017.

[29] Erich Friedman. Packing unit squares in squares: A survey and new results. *The Electronic Journal of Combinatorics*, Aug 2009. URL http://www.combinatorics.org/ojs/index.php/eljc/article/view/DS7.

[30] Boris D. Lubachevsky and Frank H. Stillinger. Geometric properties of random disk packings. *Journal of Statistical Physics*, 60(5-6):561–583, 1990. doi: 10.1007/bf01025983.

[31] Walter Mickel, Sebastian C. Kapfer, Gerd E. Schröder-Turk, and Klaus Mecke. Shortcomings of the bond orientational order parameters for the analysis of disordered particulate matter. *The Journal of Chemical Physics*, 138(4):044501, 2013. doi: 10.1063/1.4774084.

[32] Joshua A. Anderson, James Antonaglia, Jaime A. Millan, Michael Engel, and Sharon C. Glotzer. Supplemental material to: Shape and symmetry determine two-dimensional melting transitions of hard regular polygons. *Physical Review X*, 7(2), May 2017. doi: 10.1103/physrevx.7.021001. URL https://journals.aps.org/prx/abstract/10.1103/PhysRevX.7.021001#supplemental. Accessed: 2018-08-15.

[33] Mikhail A. Krivoglaz. *Theory of X-ray thermal-neutron scattering by real crystals*. Plenum Press, 1969.

[34] J. A. Zollweg, G. V. Chester, and P. W. Leung. Size-dependent properties of two-dimensional solids. *Physical Review B*, 39(13):9518–9530, Jan 1989. doi: 10.1103/physrevb.39.9518.

[35] J. Dana. Honeycutt and Hans C. Andersen. Molecular dynamics study of melting and freezing of small lennard-jones clusters. *The Journal of Physical Chemistry*, 91 (19):4950–4963, Oct 1987. doi: 10.1021/j100303a014.

[36] J. M. Häring, C. Walz, G. Szamel, and M. Fuchs. Coarse-grained density and compressibility of nonideal crystals: General theory and an application to cluster crystals. *Physical Review B*, 92(18), Sep 2015. doi: 10.1103/physrevb.92.184103.

[37] Atsuyuki Okabe. *Spatial tessellations: ; concepts and applications of Voronoi diagrams*. John Wiley Sons, 2000.

[38] Nigel Goldenfeld. *Lectures on Phase Transitions and the renormalization group*, volume 85 of *Frontiers in physics*. Addison-Wesley, 1992.

[39] Interpreted language, Jul 2018. URL https://en.wikipedia.org/w/index.php?title=Interpreted_language&oldid=849516104. Accessed: 2018-08-12.

[40] Compiled language, Jul 2018. URL https://en.wikipedia.org/w/index.php?title=Compiled_language&oldid=849318907. Accessed: 2018-08-12.