Thermodynamic and Structural Phase Behavior of Colloidal and Nanoparticle Systems

by

Michael Eric Irrgang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Materials Science and Engineering)
in The University of Michigan
2016

Doctoral Committee:

Professor Sharon C. Glotzer, Chair
Assistant Research Scientist Dr. Michael M. Engel
Professor Joanna M. Millunchick
Professor Robert M. Ziff

For Lloyd,
who woulda been here if he coulda;
and for Lin,
who's been here the whole way.

ACKNOWLEDGEMENTS

I want to thank my committee, Bob Ziff, Joanna Millunchick, Michael Engel, and especially my research advisor Sharon Glotzer, for their time, energy, and mentorship. Additionally, this research would not have been possible without my coauthors Joshua Anderson, Michael Engel, David Kofke, Andrew Schultz, and Xingchen Ye. Many post-doctoral researchers and research scientists have served as mentors to me during their time with the Glotzer group and I especially want to thank Joshua Anderson, Michael Engel, Julia Dschemuchadse, Jens Glaser, Daphne Klotsa, and Greg van Anders for all of the guidance and help they have given. The Glotzer lab has provided opportunities for amazing discussions with too many brilliant fellow students to name here, but I must make room to acknowledge the enormous contributions to my research and education by Ryan Marson and Benjamin Schultz. Thank you, guys. The Comet cluster at U.C. San Diego, used for parts of this work, is part of the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575, award DMR 140129. The Titan cluster used for parts of this work is managed by Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE- AC05-00OR22725. Additional computing resources used are maintained by ARC-TS at University of Michigan. Funding for parts of this work has come from the

DOD/ASD(R&E) under Award N00244-09-1-0062 and the National Science Foundation, Division of Materials Research Award DMR 1409620.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

CHAPTER I

Introduction

Applications of nanotechnology span diverse engineering targets including medical devices, drug delivery mechanisms, energy conversion and storage, and reconfigurable or self-healing materials. Microelectronics, optics, photonics, terahertz wireless communication and self-cleaning (omniphobic) surfaces also rely on physics occurring at length scales measured in nanometers or microns. Various lithographic and micromachining techniques can be used to produce tiny features on objects of limited size under very controlled circumstances, but engineers also pursue fabrication through the self-assembly of hierarchical materials from simpler small building blocks.

By fabricating huge numbers of particles that put themselves together, we can produce large-scale hierarchical materials. One need only look in the mirror (perhaps while holding a biochemistry book) to know it is possible to achieve great complexity from simple building blocks. Historically, we know such reproducible complexity can take a lot of trial and error and the emergent behaviors of many types of interactions across many length scales.

To produce new technologies on human time scales, we must discover design principles that we can apply to achieve target structures. Experimentalists continually develop new synthesis and processing methods to produce ever more varied nanopar-

ticles and colloids but the soft-matter community is still in the relatively early stages of developing design principles that can be applied to engineer particles for target structures or applications. At the same time, many fascinating structures are observed in laboratory situations that lack reproducibility or tunability due to poor understanding of the driving forces of these particles and the thermodynamics of these systems. Design principles are developed through application of theoretical work, analysis of experiments, and, increasingly, systematic simulations and computational investigations.

Simulators benefit from the availability of ever increasing computing power only when we have software that can take advantage of it. Engineers benefit from simulation only when the results can be related to experimentally accessible parameters. Much of my work has been devoted to improvements to our simulation toolkit and the ability to translate our results to experimental observables and tunables. My research has focused on minimal interaction complexity, but rigorous thermodynamic and structural characterization to help us know clearly what aspects of particles lead to which self-assembly characteristics.

In this dissertation, I present several new algorithms and computational tools that I helped develop to study the thermodynamics and structure of systems of mesoscale particles. The parallel Monte Carlo (MC) code discussed in chapter II is a significant advancement in simulation software for hard particle systems. There, my coauthors and I discuss the development and implementation of highly parallel hard particle thermodynamic Monte Carlo simulation code. The code is optimized to run on many CPUs or many GPUs. The hard particle Monte Carlo (HPMC) simulator supports a wide variety of shape classes, including spheres / disks, unions of spheres, sphe-

rocylinders, convex polygons, convex spheropolygons, concave polygons, ellipsoids / ellipses, convex polyhedra, convex spheropolyhedra, spheres cut by planes, and concave polyhedra. The code may be easily extended to support additional shapes or specific shapes may be optimized where currently enabled by more general overlap detection algorithms. Our work is accepted for publication (with revisions) in the journal *Computer Physics Communications*.

I present an algorithm and method for measuring pressure in hard particle MC simulations in chapter III. The state function provides important information for thermodynamic integration to compare free energies between systems and for investigating phase changes. I discuss challenges and solutions for measuring pressure efficiently in hard particle simulations. The resulting tools will be released with HPMC.

In chapter IV, I primarily provide a literature review and background material for chapter V, which presents studies of the state functions of hard polyhedron fluids. Motivated by a desire to understand the state functions of polyhedral particles previously studied in the Glotzer group, we began to investigate how the interactions between particle shape and degrees of freedom could affect the pressure – volume relationship. After reviewing some analytic approaches and performing some initial simulation work, we set aside further investigation of solid phase state functions until after the maturation of the HPMC code to investigate first fluids of hard polyhedra. We have not yet resumed the project.

I present attempts to understand the state functions of polyhedra in fluid systems in chapter V. With my coauthors, I demonstrate the effectiveness of low order virial coefficients in describing the compressibility factor of fluids of hard polyhedra. We

obtain the second virial coefficient analytically from particle asphericity and use it to define an *effective sphere* with similar low-density behavior. We present a semi-analytic equation of state for hard polyhedron fluids and compare with others previously appearing in literature. My coauthors use higher-order virial coefficients — efficiently calculated with Mayer Sampling Monte Carlo — to define an exponential approximant, which I show to exhibit the best known semi-analytic characterization of hard polyhedron fluid state functions.

In chapter VI, I describe the math behind software I developed to incorporate particle form factor into nanoparticle super lattice diffraction pattern simulations. Diffraction patterns are useful to understand the symmetries and periodicities of material samples and, ideally, to confirm or refute structural models. The scattering process and diffraction measurement maps directly to taking a slice of a Fourier transform of the scattering density of the experimental sample. Diffraction images are often simulated for a known configuration of particle centers (as in simulation data) by performing a Fourier transform on a density map of coordinates. This calculation gives the structure factor associated with the diffraction, but neglects the form factor due to particle size or scattering density, which can have a strong effect on relative peak brightness in systems with more than one particle type. In colloidal or nanoparticle systems, particle form factors are additionally complicated by shape. I give some introduction into how to simulate diffraction patterns for arbitrarily mixed systems of finite-sized particles and briefly discuss software developed for this purpose.

I apply these tools and others in chapter VII, in which I analyze transmission electron microscopy (TEM) data from my coauthors and perform various simulations to

characterize and model a binary system of nanoparticles observed to self assemble into a dodecagonal quasicrystalline tiled structure. Quasicrystals have been discovered in various soft matter and nanoscale systems, but in many situations their complex geometry impedes structure solution, and the presence of defects limits the quality of experimental samples. We report a dodecagonal quasicrystal and related periodic superlattices in a binary mixture of iron-based (FePt, Fe_3O_4, $CoFe_2O_4$) and gold nanocrystals. This quasicrystal has a structure type previously unreported. We derive a structure model for the decoration of a square-triangle tiling by analysis of surface terminations and electron tomography. The twelve-fold symmetry of the quasicrystal is broken in sub-layers with only six-fold symmetry for puckered and mirror layers, resulting in partial matching rules between square and triangle tiles. Higher-dimensional lifting works well to describe the experimental quasi-lattice though allowance must be made for occasional dislocations. A hard particle model is insufficient to explain the stability of the quasicrystal despite attempts to optimize particle shape. Instead, hetero-attraction allows the quasicrystal to grow in molecular dynamics simulations.

I conclude this dissertation in chapter VIII and describe the outlook for continuing research in these subjects.

CHAPTER II

Scalable hard particle Monte Carlo and thermodynamic measurement

This chapter consists primarily of material accepted for publication in *Computer Physics Communications* [1], coauthored with Sharon C. Glotzer and principally authored by Joshua Anderson. Subsections 2.2.2 and 2.2.3 and parts of subsection 2.2.1 do not appear in the journal article.

2.1 Introduction

CPU performance hit a performance brick wall in 2005 [2], and *serial* execution performance has remained stagnant since then. Whole socket CPU performance continues to increase due to additional CPU cores and wider single instruction multiple data (SIMD) vector instruction widths. Moore's Law drives the increase in core counts, with 2-core CPUs available in 2005 increasing to 18-core CPUs in 2015. XSEDE [3] Comet is a modern commodity dual-socket CPU cluster with 24 cores per node (12 cores per CPU). This is a typical configuration for current systems; future clusters will have more cores per node. However, CPUs are not very power efficient. Graphics processing units (GPUs) have thousands of cores and can process hundreds of thousands of concurrent lightweight threads. Given a fixed power

budget, systems that use GPUs provide significantly higher performance than those with CPUs alone. For example, the GPUs on OLCF Titan provide over 90% of its performance. When Jaguar was upgraded to Titan by adding GPUs, its total peak performance increased by 10x with only a 20% increase in power usage.

Metropolis Monte Carlo (MC) simulations for off-lattice particles are usually implemented in serial. This is the most straightforward way to evolve the Markov chain, but it can achieve only a small fraction of the performance available in a single compute node and does not scale to simulations with large numbers of particles. Efficient sampling algorithms can achieve orders of magnitude better performance than Metropolis MC, such as the event chain algorithm for hard spheres [4] and general pair potentials [5]. However, it is not clear how event chain MC can be extended to hard particles with shape, which is our primary interest.

Computational scientists need general purpose simulation tools that utilize parallel CPUs and GPUs effectively. They need to run simulations of a few thousand particles as fast as possible in order to answer research questions quickly, conduct high throughput screening studies, and sample more states with a short turnaround time. Researchers also need scalable codes to complete large simulations with millions of particles, which is untenable with a serial code. There are a number of possible routes to parallelizing Metropolis MC [6]: 1) execute many independent runs in parallel to improve sampling, 2) evaluate energies in parallel for trial moves that are proposed in serial, and 3) propose multiple trial moves in parallel. Executing many independent serial runs is not helpful for large systems or those with long equilibration times. Two recent open source codes fall into the second category. CASSANDRA [7] uses OpenMP to run in parallel on the CPU and GOMC [8] uses

CUDA to parallelize on NVIDIA GPUs. Both of these tools model atomistic systems with classical potentials.

A number of works use checkerboard techniques to propose trial moves in parallel in off-lattice systems with short ranged interactions [9–13]. Heffelfinger introduced the concept [9], but found it inefficient due to high communication overhead. Ren [11] and O'Keeffe [12] improved efficiency with sequential moves in the domains. Sequential moves obey balance in serial implementations [14], but the same argument does not apply to checkerboard parallel moves. We showed in Ref. [13] that sequential moves within active checkerboard domains lead to incorrect results, as does allowing particle displacements to cross from an active domain to an inactive one as allowed in refs [9, 11, 12]. Uhlherr [10] implemented a two color asymmetric striped decomposition, proposes complex polymer conformation moves within the domains, and correctly rejected moves that cross boundaries. Kampmann [15] combined event chain MC with the parallel checkerboard scheme in a rejection free manner by reflecting trial moves off the domain walls.

Previously, we developed a general algorithm for massively parallel Metropolis Monte Carlo, implemented it for two-dimensional hard disks on the GPU [13], and used it to confirm the existence of the hexatic phase in hard disks [16]. In this paper, we present a general purpose code for MC simulations of hard shapes, HPMC. HPMC runs NVT and NpT [17,18] ensembles in 2D or 3D triclinic boxes. Additional integration schemes permit Frenkel-Ladd [17] free energy computations and implicit depletant simulations [19]. It calculates pressure in NVT simulations by volume perturbation techniques [18, 20]. HPMC supports a wide variety of shape classes, including spheres / disks, unions of spheres, convex polygons, convex spheropolygons,

```
import hoomd_script as hoomd
from hoomd_plugins import hpmc

# Read the initial condition.
hoomd.init.read_xml(filename='init.xml')

# MC integration of squares
mc=hpmc.integrate.convex_polygon(seed=10,
                                 d=0.25, a=0.3)
square=[(-0.5, -0.5), ( 0.5, -0.5),
        ( 0.5,  0.5), (-0.5,  0.5)])
mc.shape_param.set('A', vertices=square)

# Run the simulation
hoomd.run(10e3)
```

Figure 2.1: Example HPMC job script. The syntax is preliminary and may change as we reorganize components for a final release.

concave polygons, ellipsoids / ellipses, convex polyhedra, convex spheropolyhedra, spheres cut by planes, and concave polyhedra. It runs efficiently in serial, on many CPU cores, on a single GPU, and on multiple GPUs. Researchers have already used HPMC in studies of shape allophiles [21] and ellipsoids with depletants [22].

2.2 Implementation

HPMC is an extension of HOOMD-blue [23–25] using the existing file formats, data structures, scripting engine, and communication algorithms. HOOMD-blue started off as a molecular dynamics (MD) package, but its design is general enough to allow the addition of Monte Carlo moves with minimal modifications. HPMC is an `Integrator` class inside HOOMD-blue that applies MC trial moves to the particles. The code is object-oriented and extensible, and it is easy to add additional shape classes and collective moves. Adding new types of local moves is not as easy, but can be accomplished by subclassing the `Integrator` and re-implementing the main loop.

Python job scripts control HOOMD-blue execution. Users can activate HPMC integration with a few lines, and can switch back and forth between MC and MD in the same job script. Listing 2.1 shows a job script that runs a simulation of hard squares for ten thousand steps. A single "step" in HPMC is approximately n_s sweeps, the approximation is due to the parallel domain decomposition. One sweep is defined as N trial moves, where N is the number of particles in the simulation box.

2.2.1 Metropolis Monte Carlo

Correct Monte Carlo importance sampling of a thermodynamic distribution is most easily proven by satisfying the condition of *detailed balance*. The condition imposes constraints on the acceptance criterion for trial moves in relation to the probability $P(A)$ of being in a microstate A, the probability of transitioning between microstates $\pi(A \to B)$, and the underlying transition matrix α. The acceptance criterion due to Metropolis is one way to assure detailed balance when we can use our knowledge of thermodynamics to compare the relative likelihood of two microstates (such as by a Boltzmann factor) in a thermodynamic distribution f_Ψ for some thermodynamic potential Ψ. Following Frenkel and Smit [17]:

$$\text{detailed balance: } P(A)\pi(A \to B) = P(B)\pi(B \to A)$$

$$\pi(A \to B) = \alpha(A \to B)\ \mathrm{acc}(A \to B)$$

$$\text{for symmetric } \alpha,\ P(A) \times \mathrm{acc}(A \to B) = P(B) \times \mathrm{acc}(B \to A)$$

$$\frac{\mathrm{acc}(A \to B)}{\mathrm{acc}(B \to A)} = \frac{P(B)}{P(A)} = \frac{f_\Psi(B)}{f_\Psi(A)}$$

The Metropolis criterion satisfies detailed balance.

$$\mathrm{acc}(o \to n) = \begin{cases} \frac{f_\Psi(n)}{f_\Psi(o)}, & f_\Psi(n) < f_\Psi(o) \\ 1, & f_\Psi(o) \geq f_\Psi(n) \end{cases} \tag{2.1}$$

Hard particle simulations have infinite potential energy when any particles overlap and zero potential energy otherwise. Metropolis Monte Carlo [17, 26] for hard particles with shape consists of the following steps. Let $\vec{r}_i$ and q_i be the position and orientation of particle i.

1. Select a particle i at random.

2. Generate a small random trial move for that particle, resulting in a new trial configuration $\vec{r}_{\mathrm{trial}} = \vec{r}_i + \delta\vec{r}$, $q_{\mathrm{trial}} = q_i \cdot \delta q$.

3. Check for overlaps between the trial configuration and all other particles in the system.

4. Reject the trial move if there are overlaps, otherwise accept the move and set $\vec{r}_i \leftarrow \vec{r}_{\mathrm{trial}}$, $\vec{q}_i \leftarrow \vec{q}_{\mathrm{trial}}$.

The last step is a simplification of the more general Metropolis acceptance criterion [26] for hard particle systems. It offers an important opportunity for optimization: Once the first overlap is found, no further checks need to be made.

For new simulations, we follow a general rule of thumb and select the size of $\delta\vec{r}$ and δq so that an (estimated) optimal percentage of the trial moves are accepted. A simple way to measure efficiency for fluids is the diffusion rate in wall clock time units. We check with this metric for several benchmark cases, and trial move sizes associated with a 20% acceptance ratio are at or very close to peak efficiency for the

high density fluids we are interested in. The rule of thumb is not always optimal, but it is useful as researchers can trivially implement it. All benchmark results reported in this work are initially tuned to 20% acceptance, then the trial move size is fixed.

2.2.2 Sampling volume at constant pressure

For the isobaric-isothermal (NpT) ensemble, the characteristic thermodynamic potential is the Gibbs free energy. In this case, detailed balance is satisfied [Frenkel and Smit eq. 5.4.11] for a volume change from V to $V' = V + \triangle V$ by the following Metropolis acceptance scheme. The acceptance criterion is formulated using the ratio of probabilities of states before and after trial moves drawn from the relevant distribution. From Frenkel and Smit [17],

$$\begin{aligned} f(V; \mathbf{s}^N)\, dV &\propto V^N \exp(-\beta p V) \exp(-\beta U(\mathbf{s}^N; \mathbf{L}))\, dV \\ \mathrm{acc}(V \to V') &= \min(1, \exp\left(-\beta \delta U - \beta p \delta V - N \ln \frac{V'}{V}\right) \end{aligned} \tag{2.2}$$

Equation 2.2 is derived for box moves in volume, drawn from a symmetric distribution of δV. The distribution of $\ln V$ is different and for trial moves drawn from $\ln V$,

$$\begin{aligned} f(\ln V; \mathbf{s}^N)\, d(\ln V) &\propto V^{N+1} \exp(-\beta p V) \exp(-\beta U(\mathbf{s}^N; \mathbf{L}))\, d(\ln V) \\ \mathrm{acc}(\ln V \to \ln V') &= \min(1, \exp\left(-\beta \delta U - \beta p \delta V - (N+1) \ln \frac{V'}{V}\right) \end{aligned}$$

Thus it is important to consider the correct distribution for a type of move in deriving the appropriate acceptance criterion. Currently HPMC allows trial moves in volume, box edge length, and shear, and will allow additional volume moves in the future. Note, though, that due to cancellation of factors, the acceptance criterion for changes

in V or in L (a single lattice direction) are the same and can be expressed

$$\text{accept}(V \to V') = \min\left[1, \exp\left(-\beta\left[U(V') - U(V) - \frac{N}{\beta}\ln\frac{V'}{V} + p(V' - V)\right]\right)\right] \tag{2.3}$$

For a partial discussion of the derivation of (2.3) see Frenkel and Smit [27], but there are some subtleties to the appearance of the third term.

For the hard particles, U is either zero or infinity, potentially simplifying the acceptance criterion.

$$\text{accept}(V \to V_{\text{new}}) = \begin{cases} \min\left[1, \exp\left(N\ln\frac{V_{\text{new}}}{V} - \frac{p(V_{\text{new}} - V)}{kT}\right)\right] & \text{no overlaps} \\ 0 & \text{overlaps} \end{cases} \tag{2.4}$$

In the specific case of convex hard particles in a box undergoing isotropic volume changes, optimized acceptance criteria could consist of an overlap check on compression moves and comparison of a random number to the calculated pseudo-Boltzmann factor (2.4) for expansion moves. Neither criterion is sufficient for general simulations, though, even of hard particles.

Expansion moves are just coordinate rescalings and thus never result in overlaps if performed isotropically in systems of convex particles. However, HPMC is capable of simulating concave objects and performing anisotropic NpT. That is, trial moves can be performed in box shear or edge length, so an overlap check must generally be performed for all NpT move types.

For compression moves in hard particle systems, equation Equation 2.3 is dominated by the potential energy term and the sum of the remaining terms is generally less than zero. The maximum allowable volume change for this condition to be true

depends on the compressibility factor, Z (Equation 5.1), but for physical systems where generally $Z > 1$ the condition is easily satisfied for volume changes on the order of one percent at a time. In other words, for the overlap condition to be insufficient for compression acceptance, the exponential term in (2.4) would have to be less than one, or $v_r > \exp(v_r Z) - 1, v_r \equiv \frac{\delta V}{V}$. Numerically solving for the maximum volume change from near equilibrium gives $v_r = -17.6\%$ for $Z = 1.1$ and $v_r = -94\%$ for $Z = 3$, so the condition is useful for liquid and solid systems and even for dilute systems approaching the ideal gas limit for $|v_r|$ on the order of $< 10\%$. Nevertheless, for thermodynamic integration we may want to sample near the ideal gas limit. Intuitively, we recognize that some compression moves must be rejected despite the lack of overlaps. The calculation of the pseudo-Boltzmann factor and generation of the random number are computationally inexpensive compared to the overlap check, so we disregard this potential optimization.

Pressure and temperature are not obviously separable in hard particle simulations. The HPMC NpT updater accordingly takes βp as input in a single quantity with dimension of inverse volume in simulation distance units. Users are left to adopt a reduced pressure convention of their choosing, for which one must choose a length scale relevant to the simulation at hand. E.g. $p^* = \beta P v_0$ for a monodisperse system of particles with volume v_0, $p^* = \beta P \sigma^3$ for hard sphere studies, or $p^* = \beta P a^3$ for a system of polyhedra with edge length a. Alternatively, note that $Z(\beta p, \rho)$ is independent of particle size for a given packing fraction and is thus of general utility.

Additional consideration for detailed balance

It is clear from the derivation of Equation 2.4 that detailed balance is maintained for simulations in which particle trial moves are performed in scaled coordinates,

or fractional coordinates in HOOMD parlance. Since HOOMD particle coordinates and translations are in Cartesian coordinates, to maintain strict detailed balance the Monte Carlo trial move domain would need to have the same distortions applied as the box volume. This would mean drawing translation moves from a triclinic domain that is adjusted after successful box changes.

This is needlessly complicated, however, since reversibility is regained over several MC steps if box and particle moves occur in random sequence. In the event a box move makes it impossible for a particle to undo a translation from the standpoint of relative coordinates, it is equally likely the box move will be undone before another particle move is attempted or that the particle move is undone before the box change is attempted. This argument follows the same reasoning discussed in [14].

2.2.3 Frenkel-Ladd

Free energy calculations in solid phases can be performed using Frenkel-Ladd integration [17], the implementation of which has been proved in concept for HPMC with a modified `Integrator`. Lattice sites for an Einstein crystal, along with a spring constant, are then provided as additional parameters and, in addition to the hard particle overlap checks, a Boltzmann factor must be evaluated for particle moves. The `Integrator` must log the energy of the harmonic lattice bonds at a range of spring constant values sufficient to integrate from the Einstein crystal (in which particles are sufficiently confined to their lattice sites as to never collide) to effectively zero (i.e. the simulation results are identical to those with only hard particle interactions).

Future HPMC `Integrator`s will support evaluation of non-hard-particle Boltzmann factors by optionally attaching a HOOMD `Compute` object to an `Integrator`. The idea is to provide a generic framework for external fields, including confinement,

implicit depletants, or harmonic wells.

One potential pitfall in some systems is the possibility of particles exchanging lattice sites and getting kinetically trapped for much longer than the period normally assumed sufficient for ergodicity. It has been pointed out [28] that free energy integrated from the Einstein crystal may be better sampled by treating the lattice sites as a field of (locally) harmonic wells, avoiding constraints on the mobility of vacancies. Moreover, such an approach allows particles to swap sites with less impact on the ergodicity of spring energy. However, this requires that a reasonable lattice can be constructed as input, whereas it is often preferable to choose sites arbitrarily or to use sites estimated from simulation data. Particle positions are averaged over many steps in a system sufficiently dense that particles are confined to their lattice sites. Lattice contribution to free energy can be calculated analytically independently of the simulation. Because of the high spring constant necessary to prevent particles from colliding in a dense system, it may be preferable to first thermodynamically integrate the Einstein crystal from low density to the target density and only then to integrate over the spring constant to zero.

2.2.4 Acceleration structures

A naïve implementation of hard particle MC would check $N - 1$ particles for possible overlaps with each trial configuration. The cost of a single sweep would be prohibitively slow: $O(N^2)$. Acceleration structures are data structures that reduce the execution time by efficiently identifying a subset of the N particles that possibly overlap with the trial configuration. Cell lists place particles in cells and have constant lookup time to find possible overlaps: $O(N)$ sweep execution time. Bounding volume hierarchies (BVH) build a binary tree of nodes that contain particles

and have logarithmic lookup time: $O(N \log(N))$ sweep execution time. HPMC uses cell lists on the GPU and BVHs on the CPU. Acceleration structures in HPMC are discussed in [1].

2.2.5 Overlap checks

HPMC supports many different classes of shapes. It calls the shape overlap check from the innermost loop and executes it billions of times per second in a typical simulation run, so heavily optimized shape overlap checks are needed for good overall performance. We write each shape overlap check ourselves and do not use existing libraries that would require costly data conversions and function call overhead at every check, and which lack GPU support. There is a single MC integration loop that is templated on the shape class to enable the best performance and to make code maintenance easy. Only that single class needs to be modified when fixing bugs or adding additional features to the main loop. By template instantiation, the compiler is able to inline every overlap check call, we can arrange the data in the best format for the computation, and we can choose the best overlap detection algorithm for each shape class. Users only need to write an overlap check to add a new shape class.

There are a number of methods to determine if two shapes overlap. Some methods are specific for a single class of shapes, while other are more general. When there are multiple algorithms to choose from, we test them and select the one with the best performance. For spheres and disks, overlap detection is trivial. For unions of spheres, all spheres in one shape are exhaustively checked against all those in the other. We use the separating planes method [29] for convex polygons on the CPU, but XenoCollide [30] is faster on the GPU. XenoCollide is a general algorithm that can detect overlaps between any two convex shapes. HPMC uses XenoCollide for

convex polygons, spheropolygons, convex polyhedra, convex spheropolyhedra, and spheres cut by planes. To detect overlaps between two concave polygons, HPMC checks all pairs of edges and all vertices. If no edges intersect and no vertex from one shape is inside the other, then the shapes do not overlap. We use a matrix method [22, 31, 32] to detect overlaps of ellipsoids and ellipses.

It would be expensive in both memory and compute to keep all particle geometry (e.g. polyhedron vertices) in world coordinates. HPMC efficiently represents each particle with a position $\vec{r}_i$ and an orientation quaternion q_i. Together, these describe how to rotate and then translate from the frame of the particle to the world frame. The user specifies the shape geometry in particle local coordinates once for each type of particle.

When performing an overlap check between particles A and B, HPMC works in a local coordinate system centered on particle A. The application of this local coordinate system optimization depends on the shape overlap check algorithm. For example, a support function is evaluated at every iteration of XenoCollide. The support function for the Minkowski difference $B - A$ is

$$\mathbf{S}^{\text{world}}_{B-A}(\vec{n}) = \mathbf{S}^{\text{world}}_{B}(\vec{n}) - \mathbf{S}^{\text{world}}_{A}(-\vec{n}) \; . \tag{2.5}$$

Replacing S^{world}_A and S^{world}_B with operations on the particle support functions in their local coordinate systems gives [30]

$$\mathbf{S}_{B-A}(\vec{n}) = \mathbf{R}\mathbf{S}_B(\mathbf{R}^{-1}\vec{n}) + (\vec{r}_B - \vec{r}_A) - \mathbf{S}_A(-\vec{n}) \tag{2.6}$$

where $\mathbf{R}$ is the rotation matrix that takes B into the coordinate system of A. $\mathbf{S}_{B-A}$ is in the coordinate system of particle A, but this is irrelevant for the overlap calculation. On the GPU, we replace $\mathbf{R}$ with an operation that rotates vectors by the

quaternion $q = q_A^* q_B$ because it is faster. The quaternion rotation uses more floating point operations, but requires fewer registers.

Single precision particle coordinates are not accurate in large simulation boxes (without cell-local coordinate systems [13]), so in HPMC we use double precision particle coordinates. In mixed precision mode, we compute the displacement between particles $\vec{r}_{AB} = \text{min\_image}(\vec{r}_B - \vec{r}_A)$ in double precision, then we cast $\vec{r}_{AB}$ to single precision and compute the overlap check in single precision. Within the local coordinate system of particle A, single precision is accurate for self-assembly simulations, though densest packing calculations may require full double precision. HPMC supports both full double precision and mixed precision modes as a compile time option. Full single precision builds do not even pass simple validation tests.

All benchmark and validation studies in this work use mixed precision.

2.2.6 SIMD vectorization

The polygon, spheropolygon, polyhedron, and spheropolyhedron overlap checks evaluate the support function many times in the innermost loop of XenoCollide. The support function loops over all vertices in the shape, dots them with $\vec{n}$, and returns the vertex that gives the maximum dot product. In our initial implementation, this code used over 80% of the CPU time (determined by line level profiling with `oprofile`). We improve performance of this loop with single instruction, multiple data (SIMD) vector instruction intrinsics available in modern CPUs. The first loop computes the dot products for all vertices, w vertices per iteration with SIMD parallelism, and just stores the result to avoid branch mispredication penalties around the floating point operations. A second w width SIMD loop starts and each iteration uses masks and the `BSF` assembly instruction to find the index of the maximum

element. We implement these loops in SSE ($w = 4$) and AVX (advanced vector extensions) ($w = 8$). SIMD vectorization boosts performance of the support function evaluation by a factor of 2-3 over a serial implementation with manually unrolled loops, achieving near peak floating point throughput in a microbenchmark. While the vectorized support function now executes several times faster, it is only one part of a production simulation run. We used `oprofile` to run a line level execution profile of a typical polyhedra simulation in the final version of the code. About 40 percent of the runtime is spent in the vectorized support function, 10 percent in XenoCollide iteration logic, 40 percent in AABB tree searches and the remaining 10 percent in trial moves and AABB tree generation.

2.2.7 Parallelization

Even with fast BVH trees and SIMD vector optimizations, a serial CPU simulation still only uses a fraction of the capabilities of a single compute node. We implement parallel computations that utilize the full capabilities of multi-core CPUs and clusters of CPU nodes to provide faster time to solution and to enable larger scale simulations across many nodes. In hard particle MC, there are typically only a few dozen possible overlaps with each trial configuration. This is not large enough to parallelize over a whole node and cannot scale to large simulations. The only path to achieving fast, scalable simulations for MC with short range particle interactions is to perform many trial moves in parallel [6].

To do this, we need to be able to efficiently generate many parallel random number streams. As we have before [13, 33], we use a hash based RNG, Saru [34]. Each time a trial move is generated, we hash together the particle index, time step, user seed, and MPI rank to initialize an independent RNG stream. We then use that stream

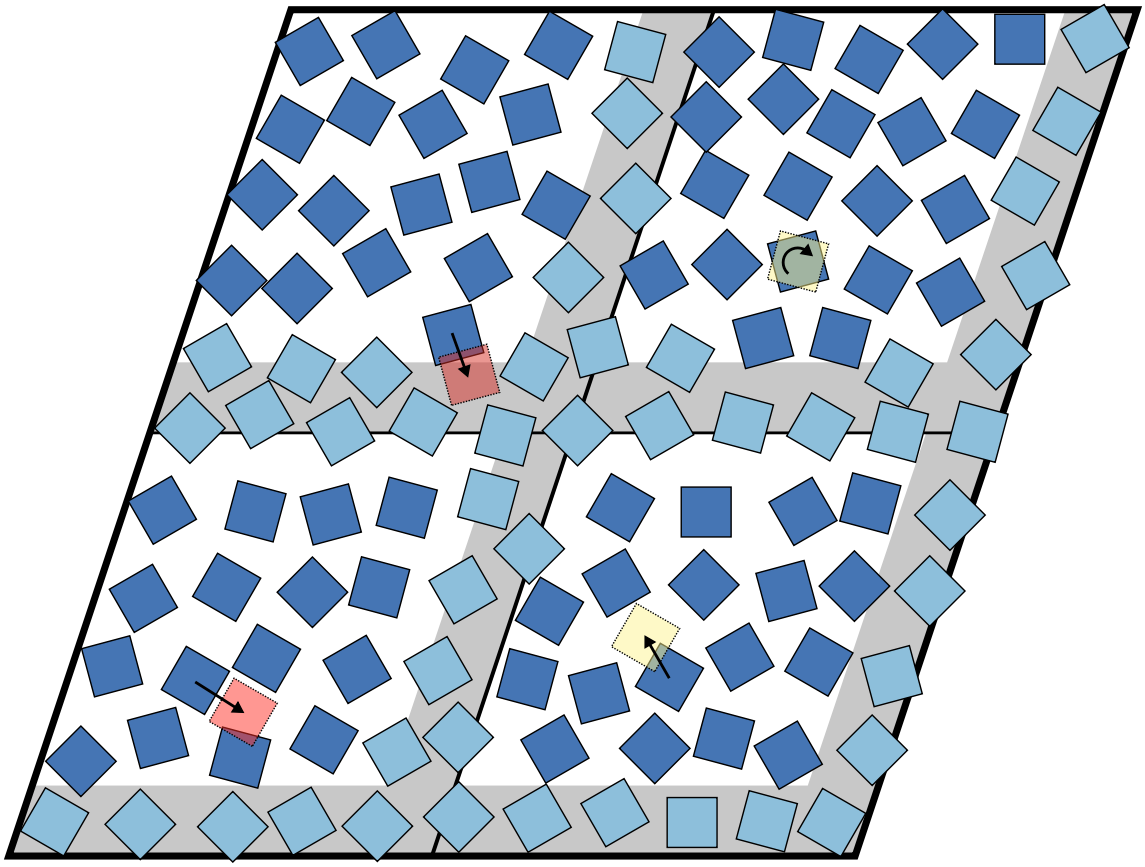


Figure 2.2: Domain decomposition scheme. The outer box is the triclinic simulation box, which is split into 4 domains. On the right and bottom edge of each domain is a gray inactive area, one particle diameter wide. Particles in the inactive region are colored lighter and are not selected for trial moves. Any trial configuration that ends in the inactive region must be rejected (top left domain in this example). On the GPU, individual domains are further subdivided with a checkerboard grid.

to generate as many random numbers as needed for the trial move.

Domain decomposition

To scale beyond a single compute node, we employ a domain decomposition strategy using MPI with one rank per CPU core, or one rank per GPU. We implement HPMC as an extension of HOOMD-blue, a parallel MD code that already has the necessary decomposition and communications routines [24]. Each rank covers a portion of the simulation box and owns all of the particles in that region. The communications routines copy particle data from neighboring ranks in a ghost layer around each domain, and migrate particles from one domain to another as they move.

We base HPMC domain decomposition on our previous method for massive parallelism [13]. However, we do not use a 2^d color checkerboard grid to scale across domains. Updating only $\frac{1}{2^d}$ of the system at a time is unnecessary with low thread counts, and would require ghost communication after every fractional system update. Instead, we modify the checkerboard scheme to have only two regions (active and inactive) and make the active region as large as possible, see figure 2.2. The inactive region has width $d_{\max}$, and it is placed along the bottom, right, and back faces of each domain.

In this layout, all inactive particles are in the neighboring domain's ghost layer, or separated from the neighboring domain's active particles by an inactive region. There is no need to communicate ghost layer updates because these particles do not move during substeps. Communication between ranks only occurs at the end of the step, when we apply a single random displacement to all particles and call the migration routine. The user sets n_{s}, the number of substeps to perform per step, giving them control of the computation to communication ratio.

In our previous work [13], we showed that shuffling the order of particles selected for trial moves achieves detailed balance within the checkerboard scheme. We proposed full shuffling of all 0–4 particle indices within a cell, as forward and reverse permutations occur with equal probability. Full shuffling causes cache thrashing in a general CPU domain decomposition implementation, where individual domains might have thousands of particles. In HPMC, we choose randomly to loop through particles either in forward or reverse index order. Both orders are cache friendly, and this selection preserves the essential element required for detailed balance: that forward and reverse sequences occur with equal probability. With this slight modification, this scheme obeys detailed balance following the same arguments as in ref. [13].

Putting all of these elements together, HPMC with domain decomposition on the CPU has the following stages.

1. Generate the AABB tree.
2. Choose forward or reverse index order randomly.
3. Loop through all particles i in the chosen order, skipping those where $\vec{r}_i$ is in an inactive region.
4. Generate a small random trial move for particle i, resulting in a new trial configuration $\vec{r}_{\text{trial}} = \vec{r}_i + \delta\vec{r}$, $q_{\text{trial}} = q_i \cdot \delta q$. Reject the trial move if $\vec{r}_{\text{trial}}$ is in an inactive region.
5. Check for overlaps between the trial configuration and all other particles in the system, using the AABB tree.
6. Reject the trial move if there are overlaps, otherwise accept the move and set

$\vec{r}_i \leftarrow \vec{r}_{\text{trial}}$, $\vec{q}_i \leftarrow \vec{q}_{\text{trial}}$. Also, update the AABB tree with the new position of particle i which may or may not require expanding its leaf node and all parents up to the root.

7. Repeat stages 2–6 n_{s} times.

8. Choose a random displacement vector and translate all particles by this vector.

9. Migrate particles to new domains and communicate ghost particles.

Stages 1–9 implement one *step*, and typical MC simulation runs continue for tens of millions of steps. The amount of useful work done by a step is proportional to the number of trial moves attempted and simulation effort is usually measured in sweeps (N trial moves). When running on a single rank, one step executes n_{s} sweeps. The ratio of active to inactive particles decreases as the number of parallel domains increases, so the number of sweeps in a step varies depending on the run configuration. Users need to be aware of this behavior so that they can configure their run protocols properly.

GPU kernel

For multi-GPU simulations, we use the same domain decomposition strategy as on the CPU but assign each active domain to a single GPU. On the GPU, we run a kernel that implements the checkerboard update scheme, similar to the one we previously implemented [13] but with a few differences. In HPMC, user configuration choices can lead to hundreds of particles in a cell, so we keep particle positions in global memory and each kernel call only proposes one trial move per cell. For disk simulations, this is slower than our specialized implementation [13], but it is not a bottleneck for complex shaped particles where the overlap check costs dominate and

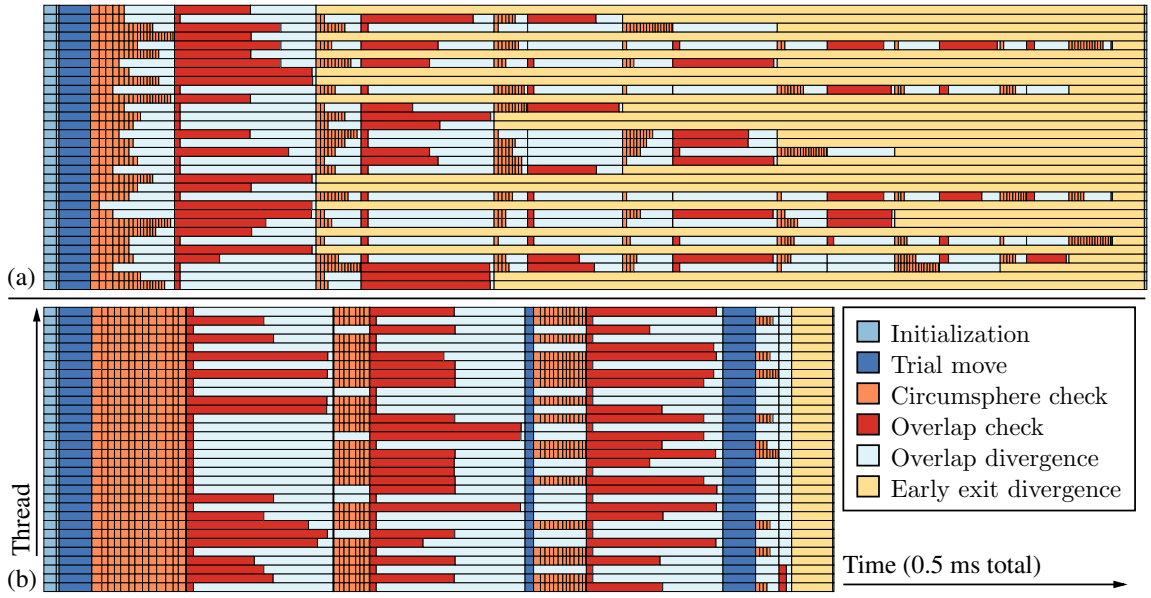


Figure 2.3: Traces of warp execution from a benchmark run of truncated octahedra. Timing data is captured with the `clock64()` function for just a single warp on the device. Colors indicate time spent in different parts of the execution. Panel (a) shows the register queue implementation and (b) shows the block queue. In panel (b), the later trial move and early exit condition rectangles indicate synchronization with other warps in the block.

accessing global memory is almost free in comparison.

To assign threads to cells, we pre-compute arrays that list the active cells for each color of the checkerboard. Then we launch 1D indexed kernels that read their cell from this array so that a single kernel may work for all use-cases. Most research relevant simulations are dense enough that the fraction of empty cells is small, though these could be removed from the list with an additional overhead per step. This structure makes one trial move for each cell that has a non-zero number of particles in it. To approach parity between a step on the GPU and a step on the CPU, HPMC uses particle density and the number of cells to estimate how many times to run the kernel so that one GPU step is approximately n_s sweeps.

Block queues

For complex shaped particles, such as polyhedra with many vertices, overlap checks take a majority of the kernel run time. They are compute limited, so GPUs have the potential to execute these checks with very high performance, but divergence is a problem. Current GPUs execute warps of 32 threads in lockstep. Threads within a warp can take different branches, but all threads in a warp execute the instructions on both sides of the branch and inactive threads are masked out. A direct translation of the MPMC checkerboard algorithm [13] loops over particles in nearby cells, checks for circumspheres that overlap with the trial configuration, and calls the full overlap check if the circumspheres overlap. In a typical simulation, there might be 100 particles in the cells around the location of a trial configuration, but only five of those pass the circumsphere test. With such low hit probabilities, that branch is likely to diverge every time, leading to a large reduction in performance.

We improve on this by changing the structure of the loop to make a register queue. Threads loop over potential neighbors, only checking the circumsphere overlap inside the loop. When a thread finds a potential overlap, it breaks out of the loop. Then the full overlap check is performed outside the loop after the threads have converged. This modification causes the overlap checks to run as converged as possible. Figure 2.3(a) shows a trace from a warp using the register queue. The next problem is immediately obvious in this figure: 80 percent of the threads end early when they find their first overlap and know that the move must be rejected. The remaining 20 percent must check all potential overlaps before accepting the move. The critical path for the entire warp to complete is determined by only 20% of the threads so divergence is still a problem.

We attempt to use a global queue to work around this. The first kernel generates trial moves, performs circumsphere checks, and inserts the needed full overlap checks into a global queue. Then a second kernel processes the queue and runs all of the overlap checks with no divergence due to circumsphere checks or early exit conditions. A third kernel applies the accepted moves. Overall, this method performs no better than the register queue kernel. It was able to compute many more overlap checks per second, but it also had to perform many more overlap checks because it is not able to take advantage of the early exit condition.

Our fastest, and final, implementation uses the idea of a work queue for the expensive overlap checks, but does so at a block level rather than at the global level. One or more threads in a group run for each cell in the active set. They generate the trial move and then loop through the particles in the nearby cells in a strided fashion. For example, with a group size of 4, thread 0 checks nearby particles 0, 4, 8, ... and thread 1 checks 1, 5, 9, ... In this phase, threads only check for circumsphere overlaps. When a particle passes the circumsphere test, the thread adds the particle index and group id to a queue in shared memory. The maximum queue size is the number of threads in the block. Once the queue is full, the loop over nearby particles exits and all threads in the block enter the next phase. Here, each thread performs the overlap check in the queue entry matching its thread index, which may be for a trial move generated by a different thread. If the particles overlap, the thread atomically increments an overlap counter for the appropriate group. Then the first phase starts populating the queue again, except that threads with already discovered overlaps do not add any work to the queue. These two phases repeat until there are no more nearby particles to check for any thread in the block, then accepted trial

moves are handled.

Figure 2.3(b) shows a trace from a warp using the block queue. The overlap check phase of the kernel runtime is kept dense and non-divergent until the last pass of the non-full queue. There is still divergence within the iterative XenoCollide overlap checks themselves. We implemented XenoCollide as a single loop to avoid divergence as much as possible, but some particle configurations require more iterations than others. We tried a variety of ways to remove overlap checks that exit in the first iteration from the queue, but adding that cost on top of every circumsphere check slowed performance overall.

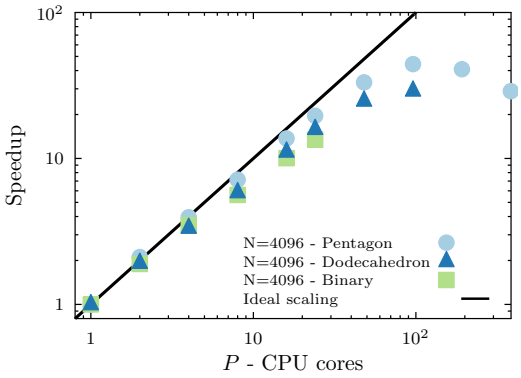
With the block queue implementation, any number of threads can be run per active cell so long as the total block size is a multiple of the warp size. This allows many threads to execute per cell, which is critical to obtain high performance on modern GPUs. HOOMD-blue autotunes kernel launch parameters to find the fastest performing values [24]. We autotune over all valid combinations of the group size and block size to find the fastest performing configuration. In cases where there are a large number of particles in nearby cells, the autotuner will pick a large group size (i.e. 8 or 32) to have many threads available to process the overlap checks. In cases where there are only a few particles in nearby cells, it chooses 1 or 2. We test a variety of benchmark cases and always find the block queue outperforms the register queue; performance benefits range from 20 to 80 percent.

2.3 Performance

We benchmark HPMC performance on a few reference systems that researchers have previously studied. Our first benchmark is a system of 2D regular pentagons in a high density fluid at a packing fraction of 0.676 in NVT. This is a single state

	Hours to run 10e6 sweeps		
P	Pentagon	Dodecahedron	Binary
1	7.56	21.78	32.94
2	3.58	11.33	17.40
4	1.91	6.52	9.48
8	1.06	3.72	5.86
16	0.55	1.96	3.28
24	0.39	1.37	2.45
48	0.23	0.88	-
96	0.17	0.75	-
192	0.19	-	-
384	0.26	-	-

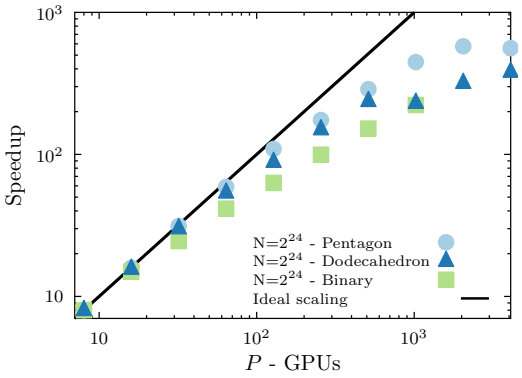
(a) Comet performance ($N = 4096$)



(b) Scaling on Comet

	Hours to run 10e6 sweeps		
P	Pentagon	Dodecahedron	Binary
8	102.03	492.83	901.73
16	51.25	252.48	482.17
32	26.04	131.01	294.12
64	13.82	73.42	173.80
128	7.47	44.61	114.48
256	4.68	26.32	72.52
512	2.83	16.64	47.45
1024	1.83	17.13	32.42
2048	1.42	12.40	-
4096	1.46	10.36	-

(c) Titan performance (N=2^{24})



(d) Scaling on Titan

Figure 2.4: Performance is reported in hours to complete 10e6 sweeps. Scaling plots are performance values renormalized to performance on the smallest P run. Simulations are run on CPUs on XSEDE Comet (Intel Xeon E5-2680v3) with $N = 4096$ and GPUS on OLCF Titan (NVIDIA Tesla K20X) with $N = 2^{24}$.

point in a previous study by Schilling, Frenkel et al. [35]. Our second benchmark is a system of 3D dodecahedra in a high density fluid at a packing fraction of 0.5 in NVT. This is representative of monodisperse self-assembly simulations of polyhedra [36]. Binary systems have a much larger phase space to explore (composition, size ratio), so such studies are computationally expensive and can benefit greatly from optimized, parallel simulation codes. Khadilkar and Escobedo [37] studied a binary mixture of tetrahedra and octahedra with equal edge lengths that could tile space (volume ratio

1 : 4). We use this system for our third benchmark, in the solid at a packing fraction of 0.6 in NVT. The binary benchmark benefits greatly from the BVH tree, though the size ratio is not large enough to demonstrate the full capabilities of the tree to efficiently simulate huge size disparities. We leave those benchmarks for other papers on methods to model systems with large colloids and small depletants [19] and MD methods using BVH trees [38].

For all three benchmarks, we explore strong scaling performance in two regimes. The first case is $N = 4096$, a system size representative of what researchers have used in previous studies with serial MC implementations. Such systems are too small to run efficiently on the GPU, but parallel CPU simulations offer tremendous speedups over serial ones. We run this case on XSEDE Comet, a recent addition to the XSEDE ecosystem. Comet has dual-socket nodes with Intel Xeon E5-2680v3 CPUs — a total of 24 cores per node. Figure 2.4(a–b) shows the performance of the three benchmarks at $N = 4096$ on Comet. Both the pentagon and dodecahedron benchmarks scale out to 96 CPU cores, only 43 particles per domain. At this point, it takes 10 minutes to run 10 million sweeps in the pentagon benchmark, and 45 minutes in the dodecahedron benchmark. Contrast that with serial simulations that would take 7.6 and 21.8 hours, respectively. Due to the size disparity, the binary benchmark does not decompose over 24 cores. Past that point, the inactive region covers the whole domain. Still, we reduce a serial runtime of 32.9 hours down to 2.45 to complete 10 million sweeps.

The second regime we benchmark is large systems of $N = 2^{24}$ (16.8 million) particles. Running such a large system is inconceivable with a serial simulation code, where it would take more than a month to complete 10 million sweeps and

years to equilibrate a system. Large systems easily fill the GPU, so we run these benchmarks on OLCF Titan, which has 1 NVIDIA Tesla K20X GPU per node. Figure 2.4(c–d) shows the results. The pentagon benchmark scales out to 2048 GPUs (8192 particles/GPU), where it takes 1.4 hours to complete 10 million sweeps. The dodecahedron benchmark scales out to 4096 GPUs (4096 particles/GPU), where it takes 10.36 hours to complete 10 million sweeps. As on the CPU, domain size limits the scaling of the binary benchmark, this time to 1024 GPUs (16384 particles/GPU).

The strong scaling limit is the fastest possible simulation one can achieve, though it uses compute resources inefficiently. Given a fixed compute time budget, one can get more simulations completed with fewer MPI ranks at the cost of longer wait times to finish each run. Efficiency depends primarily on the number of particles per CPU core (or per GPU). HPMC obtains a reasonable efficiency of 60–70% with 85 particles per CPU core for the pentagon benchmark and 170 for the dodecahedron benchmark. The same efficiency is reached at 65536 pentagons/GPU and 131072 dodecahedra/GPU, though Titan's usage policies strongly encourage runs much closer to the strong scaling limit. These are representative of 2D and 3D simulations of single particle type systems in general, so researchers can use these as rules of thumb. For systems with particle size disparity, we advise users to run their own short scaling benchmarks for their systems to determine an appropriate selection. Efficiency as a function of N/P varies greatly with size ratio and composition parameters.

GPU speedup over the CPU is not a focus of this work; we instead present what types of simulations the CPU and GPU hardware architectures are well-suited for and how well HPMC performs those benchmarks. However, some readers may still be interested in relative speedup. It is difficult to make a GPU/CPU comparison at scale

and it is not fair to compare several year old K20X GPU to the brand new Haswell CPUs on Comet. However, these are the systems currently available to researchers at scale — Comet has K80s, but limits users to no more than 16 GPUs at a time. One way to compare is to pick points at the 60–70% efficiency level and compare trial moves per second. At reasonable efficiency on the pentagon benchmark, HPMC performs 12.5 million trial moves per second per CPU socket and 38.9 million per GPU socket, for a socket to socket speedup of 3.1. At reasonable efficiency on the dodecahedron benchmark, HPMC performs 4.45 million trial moves per second per CPU socket and 8.16 million per GPU socket, for a socket to socket speedup of 1.8.

2.4 Validation testing

We rigorously test HPMC for validity at three levels. At the lowest level, we perform unit tests on the AABB tree, move generation code, and shape classes. We test that AABBs are generated properly, and that queries on the resulting trees find all possible overlapping particles. We verify that trial moves are generated from the proper uniform distribution and that the particle update order is correctly randomized. For each shape class, we place many test configurations and validate that overlapping and non-overlapping configurations are correctly detected. This is essential to ensure the quality of the overlap check algorithms as there are many corner cases to account for. The shape overlap unit tests contain many configurations captured from simulation runs that we identified were overlapping by independent methods. Low level unit tests cover 14 classes with over 1400 different checks.

MC `Integrator`s cannot be checked with low level unit tests because their stochastic nature makes it impossible to define what a correct output is given an input. Instead, we validate the `Integrator`s with system level tests which are python job

scripts that perform simulation runs. System tests verify different operating modes of the `Integrator` and ensure that documented interfaces for controlling those modes work. Additionally, we run short simulations and check for any overlaps in the generated configurations. Most bugs in the `Integrator` implementation result in accepted moves that have overlapping particles.

We run unit and system level tests on every commit to the repository with a Jenkins continuous integration server. Jenkins runs these tests on the CPU and on 4 different generations of GPU, in both mixed and double precision, and with and without MPI for a total of 20 build configurations. It e-mails developers when a commit fails any of the tests.

Unit and system level tests are designed to run quickly and automatically to detect bugs in the implementation. They are not sufficient to verify that HPMC correctly samples the ensemble of states available to the system. Such tests take much longer to run and we do so by hand. We test three separate systems to validate HPMC in NVT: disks, spheres, and truncated octahedra. We run each system in multiple compute configurations, sample the pressure to high precision, and ensure that all simulations produce the same result. For hard disks, we have three independent data points to compare to from event chain MC, event driven MD, and our previous GPU checkerboard implementation [16]. No such high precision data exists to validate hard spheres and truncated octahedra, so we instead verify that serial and all parallel builds agree. From our previous work, and from tests that introduce issues in HPMC, we know that this validation technique is sensitive enough to detect when there are subtle problems - such as looping through particles in sequence instead of randomly choosing the forward or reverse order.

Mode	Disk	Sphere	Truncated octahedron
Serial	9.1707(4)	9.3135(4)	13.8975(17)
24 CPU cores	9.1709(2)	9.3136(2)	13.8972(11)
1 GPU	9.1708(2)	9.3134(2)	13.8970(7)
4 GPUs	9.1710(3)	9.3134(2)	13.8970(6)

Table 2.1: Pressures obtained during NVT validation test runs, in reduced units. In 2D, $p^* = \beta p \sigma^2$, where σ is the diameter of the disk. In 3D, $p^* = \beta P v_0$, where the reference volume v_0 is the single particle volume for the respective shape. The numbers in parentheses are two standard errors of the mean in the last given digit(s).

Specifically, we run: 65536 hard disks at a packing fraction of 0.698 (fluid), 131072 hard spheres at a packing fraction of 0.60 (solid), and 16000 truncated octahedra at a packing fraction of 0.7 (solid). We initialize the hard disks randomly and allow them to equilibrate, while we place the two solid systems on the known self-assembled FCC and BCC lattices. We prepare a number of independent equilibrated initial configurations and run as many sampling runs, 30 for disks, and 8 for the other shapes. We ran each parallel disk simulation for 60 million sweeps (only 24 million in serial), spheres for at least 8 million, and truncated octahedra for up to 80 million sweeps, all after suitable equilibration periods. Estimated error is reported as 2 standard errors of the mean from the independent runs. We perform the set of runs in serial, on 24 CPU cores in parallel with domain decomposition, on a single GPU with checkerboard parallelism, and on 4 GPUs with both checkerboard and domain decomposition. Tests are performed on the XSEDE Comet and University of Michigan Flux systems.

To sample the pressure in NVT runs, we build a histogram of scale factors s that cause two neighboring particles to overlap and extrapolate the probability of overlap at $s = 0$. This is a generalization of the $g(r)$ based technique we previously used for hard disks [13, 16], see those references for full details on how to sample and

extrapolate the histogram without introducing any systematic errors. We use a more general volume perturbation technique [18,20] to extend this method to particles with shape.

Figure 2.1 shows the results of these tests. For each shape, all run configurations give the same pressure within error and verify that HPMC performs correct simulations in all parallel modes. Additionally, HPMC's hard disk results match, within error, the previous pressures obtained with three simulation methods [16], where $P^* = 9.1707(2)$ for this state point.

2.5 Conclusions

We presented HPMC, a parallel simulation engine for hard particle Monte Carlo simulations we developed as an extension to HOOMD-blue. HPMC executes in parallel on many CPU cores and many GPUs, and we optimized HPMC to run as fast as possible on both architectures. On the CPU, we used efficient bounding volume hierarchies to search for possible overlaps, and SIMD vector intrinsics in the innermost loop to take full advantage of modern processors. On the GPU, we performed trial moves in parallel on a checkerboard with many threads per cell, and implemented a block level queue to limit performance degradation due to divergence.

Our implementation is general and works for any shape, given an implementation of an overlap check. Users can easily add new shapes to the code without needing to write GPU kernels. HPMC ships with overlap checks for many classes of shapes, including spheres / disks, unions of spheres, convex polygons, convex spheropolygons, concave polygons, ellipsoids / ellipses, convex polyhedra, convex spheropolyhedra, spheres cut by planes, and concave polyhedra.

Completing 10 million sweeps of a system of 4096 pentagons required 7.6 hours

in serial. HPMC achieved the same in 10 minutes when running in parallel on 96 cores. GPUs allow efficient runs with tens of millions of particles. On 2048 GPUs, HPMC ran 10 million sweeps of a system of 16.8 million pentagons in 1.4 hours.

HPMC is available open-source in HOOMD-blue, starting with version 2.0.

Acknowledgments

HPMC is a group effort. We acknowledge the following group members for contributions to various parts of the code, Khalid Ahmed, Michael Engel, Jens Glaser, Eric S. Harper, and Benjamin A. Schultz.

Parts of this work were supported by the DOD/ASD(R&E) under Award No. N00244-09-1-0062 (initial design and implementation) and the National Science Foundation, Division of Materials Research Award # DMR 1409620 (BVH tree implementation and GPU kernel optimizations). Software was validated and benchmarked on the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575, on resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725, and was also supported through computational resources and services provided by Advanced Research Computing Technology Services at the University of Michigan, Ann Arbor.

The Glotzer Group at the University of Michigan is an NVIDA GPU Research Center. Hardware support by NVIDIA Corp. is gratefully acknowledged.

CHAPTER III

Measuring pressure in hard particle simulations

This chapter includes material being prepared for submission to *Molecular Physics* with coauthors Michael Engel, Joshua A. Anderson, and Sharon C. Glotzer.

3.1 Introduction

In the field of colloidally self-assembled materials, simulations of large systems allow researchers to study phase coexistence and nucleation, or to observe novel phenomena such as hexatic phases. One of the basic tools for investigating the thermodynamics of any system is to measure state functions. However, there are computational challenges for the non-differentiable potentials of the simplest particle models, hard colloids, whose interaction is by excluded volume only and dominated by the particle geometry.

In particular, simulations in the constant pressure (NpT) ensemble are slow. In general, the simulation box is one of the slowest observables to decorrelate in hard particle Monte Carlo. Additionally, unlike simulations at constant volume (NVT), the time to convergence for NpT simulation is super-linear in system size, rendering NpT simulations practically impossible in many cases. The tractability of an NpT simulation is also ultimately limited by the precision of the simulation to make box

changes sufficiently small to succeed. Pressure measurement in NVT simulations is analytic for few particle shapes. Perturbative numerical techniques, historically underutilized, are unclear or insufficient as previously published [18, 20, 39–41] for general application to large systems of arbitrary particle shapes.

In this contribution, we combine and extend previous methods of pressure calculation in hard particle simulations from a perturbative thermodynamics approach and present an efficient and highly scalable implementation that works for hard particles of any shape. This has allowed us to extract previously elusive thermodynamic data with high computational efficiency. We implement our method in a highly parallel hard particle Monte Carlo code for CPU and GPU and demonstrate its use.

3.2 Background

In 1950, Kirkwood [42] summed up existing theory relating intermolecular potential, radial distribution, and other thermodynamic functions of liquids, noting the equation of state for a liquid with pair-wise isotropic intermolecular force. The time-averaged Clausius virial function, separated into external (applied pressure) and internal (particle interaction) terms gives the familiar virial equation:

$$\frac{\beta p}{\rho} = 1 - \frac{\beta}{\nu N} \left\langle \sum_{i=1}^{N} \mathbf{r}_i \cdot \nabla_i \phi_i(\mathbf{r}^N) \right\rangle \tag{3.1}$$

for the N particles in ν dimensions and potential ϕ. We commonly assume ergodicity to equate the time average and ensemble average in Equation 3.1, which is the formula commonly used to calculate pressure in molecular dynamics simulations [17]. An equation of state for hard spheres is then straightforward [43] in terms of the radial pair distribution function, a quantity measurable in NVT simulations.

$$\frac{\beta p}{\rho} = 1 - \frac{2\pi\beta\rho}{3} \int_0^\infty dr\, \phi'(r)\, g(r)\, r^3$$

Let $y = \exp(\beta\phi)g$, $e = \exp(-\beta\phi) \rightarrow e' = -\beta\phi' \exp(-\beta\phi)$. Then, for hard spheres of diameter σ,

$$\begin{aligned}\frac{\beta p}{\rho} &= 1 + \frac{2\pi\rho}{3}\int dr\, r^3 y(r) e'(r) \\ &= 1 + \frac{2\pi\rho}{3}\int dr\, r^3 y(r)\delta(r-\sigma),\end{aligned}$$

which gives the well known result for pressure in a system of hard spheres.

$$\frac{\beta p}{\rho} = 1 + \frac{2\pi\rho}{3}\sigma^3 g(\sigma^+) \tag{3.2}$$

where $g(\sigma^+)$ is taken to be the limit of $g(r)$ from the positive direction approaching contact.

The approach can be generalized using a contact function, if one can be found [44, 45], opening the way for pressure measurement in NVT simulations using statistics across many particle pairs. More generally, the equation of state for hard particles can be expressed in terms of the probability of introducing overlaps between particle pairs under small volume perturbation.

For an isolated system (closed and thermally insulated) that does work by changing volume adiabatically and quasi-statically, conservation of energy (first law of thermodynamics) tells us that the internal energy U of the system must change as $\left(\frac{\partial U}{\partial V}\right)_{N,S} \mathrm{d}V = -p\mathrm{d}V$. The thermodynamic potential of the NVT ensemble is the Helmholtz free energy, here denoted F, so by Legendre transform $p = -\left(\frac{\partial F}{\partial V}\right)_{N,T}$. This serves as the starting point for the so-called "thermodynamic route" to pressure measurement by volume perturbation taken by several authors [18, 20, 39–41]. The thermodynamic route also leads naturally to measurement of the full pressure tensor, even in systems of concave particles.

Eppenga and Frenkel [20] introduced the idea of per-particle volume perturbations to calculate pressure, demonstrating it leads to the same results as the virial approach. The technique was later treated by Harismiadis, Vorholz, and Panagiotopoulos [39] in application to potentials with non-infinite energies (continuous potentials) at fixed ghost volume change. Evaluating the expectation value of the Boltzmann factor at fixed volume change is reasonable for finite-energy systems, but projecting to zero volume change is useful and accurate, as well as necessary, for hard particle systems.

A less terse general description incorporating volume increases and decreases appears in [40], drawing attention to interpretation of the approximation as a finite difference method. The expression used, based on a central finite difference derivative of free energy in volume, is valid for continuous potentials but not for discontinuous potentials (with a non-differentiable free energy volume derivative).

Eppenga and Frenkel do not generalize the technique by considering volume expansions as must be for full pressure tensors for anisotropic particles [18]. They consider only the scalar pressure of convex hard particles.

Brumby, *et al.*, give a clear and thorough review of the perturbative thermodynamic route to calculation of the full pressure tensor for arbitrary particles (convex, concave, hard or soft). However, they do not provide a clear route to using statistics from the overlap state of every particle individually (as per Eppenga and Frenkel), if such was their intent, rather than the state of the system as a whole. Additionally, the histogramming method for determining probabilities can be improved upon. It remains unclear why it appears to be uncommon to perform pressure calculation by per-particle contributions to free energy changes under volume perturbation.

3.3 Theory

3.3.1 Thermodynamic route

Summarizing [18, 20, 39–41], the thermodynamic definition of pressure allows us to quickly find an equation of state in terms of an ideal gas term and an excess term as a derivative. The forward difference equation leads us to an expression in terms of the expectation value of the ratio of Boltzmann factors between microstates for a system undergoing a small volume perturbation.

$$p = -\frac{\partial F}{\partial V} = -\frac{\partial}{\partial V}F_{\text{ideal}} - \frac{\partial}{\partial V}F_{\text{excess}} \tag{3.3}$$

$$= \frac{\partial}{\partial V}k_{\text{B}}T\ln\mathcal{Z}_{\text{id}} + \frac{\partial}{\partial V}F_{\text{excess}} \tag{3.4}$$

$$\beta p = \rho + \frac{1}{\beta}\lim_{\delta V\to 0}\frac{F' - F}{\delta V} = \rho + \frac{1}{\beta}\lim_{\delta V\to 0}\frac{k_{\text{B}}T\ln\mathcal{Z}' - k_{\text{B}}T\ln\mathcal{Z}}{\delta V} \tag{3.5}$$

$$\beta p = \rho + \lim_{\delta V\to 0}\frac{\ln\left\langle\left(1+\frac{\delta V}{V}\right)^N\exp(-\beta\delta U)\right\rangle}{\delta V} \tag{3.6}$$

where the expectation value is an ensemble average.

For hard particle systems, $\exp(-\beta\delta U)$ is 0 or 1 depending on whether the change introduces overlaps into the system. For $P_{\text{NO}}(\delta V)$ the probability of no overlaps being generated by the volume change δV,

$$\beta p = \rho + \lim_{\delta V\to 0}\frac{\ln P_{\text{NO}}}{\delta V}. \tag{3.7}$$

3.3.2 Detailed balance route

Recall the discussion of Equation 2.3 and Equation 2.4. Then, given the satisfaction of detailed balance, we can equate the compression and expansion acceptance probabilities. For the probability P_{NO} of no overlaps in the system resulting from a

volume change, we have

$$P_{\mathrm{NO}}(\delta V) = \exp\left(N \ln \frac{\delta V + V}{V} - \beta p \delta V\right) \tag{3.8}$$

$$\ln P_{\mathrm{NO}}(\delta V) = N \ln \frac{\delta V + V}{V} - \beta p \delta V. \tag{3.9}$$

Continuing to assume small δV,

$$\lim_{\delta V \to 0} \frac{\ln P_{\mathrm{NO}}(\delta V)}{\delta V} \to \frac{N}{V} - \beta p \tag{3.10}$$

where we use $\ln(1 + \frac{\delta V}{V}) = \frac{\delta V}{V}$ to first order in δV.

Observe that this can be rearranged into a familiar equation of state by multiplying both sides by $-V/N$ and adding 1 to both sides.

$$\frac{\beta p V}{\rho} = 1 - \frac{V}{N} \lim_{\delta V \to 0} \frac{\ln P_{\mathrm{NO}}(\delta V)}{\delta V} \tag{3.11}$$

Equation 3.11 is presented in terms of a density perturbation rather than a volume perturbation in [20]. The expression is left more general in [41] for application to soft potentials.

For anisotropic particles, anisotropic box changes can introduce overlaps. For concave particles, even isotropic box expansions can introduce overlaps. Equation 3.11 can be extended to treat the full pressure tensor for general hard particles [18] by arguing that, for small δV,

$$P(V \to V + \delta V) = P(V \to V - \delta V).$$

Then, $\mathrm{acc}(V \to V - \delta V) = \mathrm{acc}(V \to V + \delta V)$

$$P_{\mathrm{NO}}(V \to V - \delta V) = P_{\mathrm{NO}}(V \to V + \delta V) \exp\left(N \ln \frac{V + \delta V}{V} - \beta p \delta V\right).$$

Taking the logarithms of both sides and applying the same small δV assumptions as

before, then,

$$\frac{\beta p}{\rho} = 1 - \frac{V}{N} \lim_{\delta V \to 0} \left(\frac{\ln P_{\mathrm{NO}}(V \to V - \delta V) - \ln P_{\mathrm{NO}}(V \to V + \delta V)}{\delta V} \right). \tag{3.12}$$

Next, replace the probability of no overlaps in the system with the product of the probabilities that each particle has no overlaps. Then replace the probability of no overlaps with one minus the probability of at least one overlap, and linearize the remaining logarithm.

Observe that the probability P_{NO} of no overlaps in the system is the product of the probabilities of each of the pairs not overlapping. Also note that the probability P_{NO}^S of a single particle having no overlaps is the product of the probabilities that it does not overlap with any other particle.

$$P_{\mathrm{NO}}^S = \langle P_{NO,i} \rangle = \langle P_{i\,1} \cdot P_{i\,2} \cdots P_{i \neq j} \cdots P_{i\,N} \rangle = \left\langle \prod_{i \neq j} P_{i\,j} \right\rangle \tag{3.13}$$

then

$$\begin{aligned} P_{\mathrm{NO}} &= \left\langle \prod_{i}^{N} \prod_{j>i} P_{i\,j} \right\rangle = \prod_{i}^{N} \left\langle \prod_{j>i} P_{i\,j} \right\rangle = \prod_{i}^{N} \left\langle \sqrt{\prod_{j \neq i} P_{i\,j}} \right\rangle = \prod_{i}^{N} \left\langle (P_{NO,i})^{\frac{1}{2}} \right\rangle \\ &= \prod_{i}^{N} \langle P_{NO,i} \rangle^{\frac{1}{2}} \end{aligned} \tag{3.14}$$

$$P_{\mathrm{NO}} = \prod_{i}^{N} \left(P_{\mathrm{NO}}^S \right)^{\frac{1}{2}} = \left(P_{\mathrm{NO}}^S \right)^{\frac{N}{2}} \tag{3.15}$$

If the reader finds this probability manipulation too tenuous, follow Eppenga and Frenkel [20] to arrive at (3.15) by a different route. In both cases, it is observed that the correlation between introduced overlaps becomes vanishingly small in the limit of small volume perturbations. $P_{NO,i}$ is independent and identically distributed.

Equation 3.11 becomes

$$\begin{aligned}\frac{pV}{Nk_{\mathrm{B}}T} &= 1 - \frac{V}{N}\lim_{\delta V\to 0}\frac{\frac{N}{2}\ln P_{\mathrm{NO}}^{S}(\delta V)}{\delta V} \\ &= 1 - \frac{V}{2}\lim_{\delta V\to 0}\frac{\ln P_{\mathrm{NO}}^{S}(\delta V)}{\delta V}\end{aligned} \tag{3.16}$$

Replacing the probability of no overlaps with one minus the probability of at least one overlap, we have

$$\begin{aligned}\frac{pV}{Nk_{\mathrm{B}}T} &= 1 - \frac{V}{2}\lim_{\delta V\to 0}\frac{\ln\left(1 - P_{O}^{S}(\delta V)\right)}{\delta V} \\ &= 1 + \frac{V}{2}\lim_{\delta V\to 0}\frac{P_{O}^{S}(\delta V)}{\delta V}\end{aligned} \tag{3.17}$$

once we linearize the remaining logarithm. We see, then, that a reduced pressure $\frac{p}{kt}$ is easy to determine from within the machinery of the Monte Carlo simulation by gathering statistics of how many particle overlaps are generated by compression attempts.

If we consider a volume change resulting from a box rescaling, for which box vectors and coordinates are all rescaled as $x \to x(1+\lambda)$, then $V' = V(1+\lambda)^d$ in d dimensions and $\delta V = V' - V = V((1+\lambda)^d - 1)$. Note that $\lim_{\lambda\to 0}(1+\lambda)^d - 1 = d\lambda$

The compressibility factor Z is then given by

$$Z = \frac{\beta pV}{N} = 1 + \frac{1}{2d}\lim_{\lambda\to 0}\frac{P_{O}^{S}(\lambda)}{\lambda}. \tag{3.18}$$

for d the number of spatial dimensions of the system volume, λ a scaling factor (as for the volume change $V \to V' = V(1+\lambda)^d$), P_O^S the probability for a single particle to produce an overlap under rescaling, and thermodynamic β.

3.3.3 Overlap probability as a function of free volume and free surface

For cavity volume and surface area v_f and s_f in ν dimensions, we have that [46–48]

$$\frac{\beta p}{\rho} = 1 + \frac{\sigma}{2\nu}\left\langle \frac{s_f}{v_f} \right\rangle \tag{3.19}$$

The probability of overlap can be similarly estimated for a volume change. A compression move causes overlap if a particle sitting near the surface of its free volume moves outside of its free volume as a result of the coordinate rescaling. For a coordinate rescaling of $x \to x(1+\lambda)$, there is a skin region to the free volume of a particle with thickness $\sigma\lambda$ for which overlaps will occur. Thus,

$$P_O^S(\lambda) = \left\langle \frac{\sigma\lambda s_f}{v_f} \right\rangle = \sigma\lambda \left\langle \frac{s_f}{v_f} \right\rangle \tag{3.20}$$

Combining (3.20) and (3.18) returns (3.19).

3.3.4 Equivalence to the virial route to hard sphere pressure

For d dimensional spheres of diameter σ, the probability of producing an overlap is the probability of at least one particle being present in the spherical shell between the old excluded volume and the new. For small λ the volume of the shell is the area of a sphere of radius σ times the thickness $\lambda\sigma$. The probability of a particle being located in a volume V is the particle density $\rho(\mathbf{r})$ integrated over the volume. Note that $g(r)$ is a well-behaved function and for sufficiently small range of integration we can take it to be constant.

$$\lim_{\lambda\to 0} P_O^S(\lambda) = \lim_{\lambda\to 0}\int_V^{V'} \rho(\mathbf{r})\, d^3\mathbf{r} = \lim_{\lambda\to 0}\int_\sigma^{\lambda\sigma} A(r)\rho g(r)\, dr = A(\sigma)\lambda\sigma\rho g(\sigma^+)$$

$$\lim_{\lambda\to 0}\frac{P_O^S(\lambda)}{\lambda} = \begin{cases} 2\pi\sigma^2\rho g(\sigma^+) & d=2 \\ 4\pi\sigma^3\rho g(\sigma^+) & d=3 \end{cases}$$

Then, by Equation 3.18

$$\frac{\beta pV}{N} = \begin{cases} 1+\frac{\pi}{2}\sigma^2\rho g(\sigma^+) & d=2 \\ 1+\frac{2\pi}{3}\sigma^3\rho g(\sigma^+) & d=3 \end{cases} \tag{3.21}$$

$$\beta p = \begin{cases} \rho+\frac{\pi}{2}\sigma^2\rho^2 g(\sigma^+) & d=2 \\ \rho+\frac{2\pi}{3}\sigma^3\rho^2 g(\sigma^+) & d=3 \end{cases} \tag{3.22}$$

Here, $g(\sigma^+)$ is found by building a histogram of particle separations for $r > \sigma$, fitting a curve and projecting to $r = \sigma$. Note the equivalence to Equation 3.2.

3.4 Method

For non-spherical particles, the relationship between $g(r)$ and the probability of a collision is not as clear. Since we can, however, produce a histogram of the scaling factor required to produce collisions for each particle in the system, we can restate the problem. If the density $s(x)dx$ gives the number of particles whose first collision occurs at a scaling factor in the range $[x-\frac{\delta}{2}, x+\frac{\delta}{2})$, then the function

$$S(\lambda) \equiv \int_0^\lambda s(x)\, dx \tag{3.23}$$

is the number of particles that have had at least one collision as the system was scaled by λ. Note that $S(\lambda)$ goes to N for large λ.

As above with $g(r)$, we assert that $s(x)/\lambda$ is a well-behaved function such that for very small range of integration it can be taken to be constant.

$$\lim_{\lambda \to 0} \frac{P_O^S(\lambda)}{\lambda} = \lim_{\lambda \to 0} \frac{S(\lambda)}{\lambda N} = \lim_{\lambda \to 0} \frac{\int_0^\lambda s(x)\, dx}{\lambda N} = \frac{1}{N} s(0^+) \tag{3.24}$$

As before, $s(x)$ can be built as a histogram from which the value $s(0^+)$ extrapolated.

$$\frac{\beta p V}{N} = \begin{cases} 1 + \frac{1}{4}\frac{s(0^+)}{N} & d = 2 \\ 1 + \frac{1}{6}\frac{s(0^+)}{N} & d = 3 \end{cases} \tag{3.25}$$

$$\beta p = \begin{cases} \rho(1 + \frac{1}{4}\frac{s(0^+)}{N} & d = 2 \\ \rho(1 + \frac{1}{6}\frac{s(0^+)}{N} & d = 3 \end{cases} \tag{3.26}$$

3.4.1 Implementation

$s[i]$ is to be an unweighted histogram of the number of particles for which the first collision occurs at a scaling factor corresponding to bin i, where $i\delta \leq x_i < (i+1)\delta$ for bin width δ. When an analytic or numeric solution to the collision distance between particles is possible, contributions to the histogram can be calculated directly. A more general method is to determine the collision distance by bisection, which is $\mathcal{O}(\ln N)$ in number of bins rather than linear when all bin positions are evaluated. For concave particles or for anisotropic volume scalings, the system energy is not monotonic across the scaling parameter. That is, configurations may alternate between overlapping and non-overlapping at different bin values of the histogram, raising concern over whether the bisection technique is valid. However, the first overlapping configuration could similarly be missed by digitization of any histogramming

method. Moreover, we are encouraged by the validity of NpT simulations, which sample non-overlapping configurations regardless of whether a continuous transformation between steps would result in passage through an overlapping configuration. Future work will include further testing of histogramming methods for different systems, but here we restrict ourselves to measuring scalar pressure in systems of convex particles. Thus, for each particle in the system, we solve by bisection the scaling value x_i that causes a collision for each neighbor within a cutoff radius defined by the sum of the two circumsphere diameters and $\lambda_{\max}$. If $x_i \leq \lambda_{\max}$, we add 1 to the corresponding bin $s[i]$.

Using $x[i] = (i + \frac{1}{2})\delta$, a polynomial can be fit to the points $s[i]$ and $x[i]$ and solved for $s(\lambda \to 0)$. A fifth order polynomial seems to yield a low error extrapolation. It has been pointed out that a linear fit in the logarithm of probability versus density change may also be effective [20] but brief experimentation indicated the precision lost by this unnecessary simplification could limit precision to as few as three significant digits.

Furthermore, using $g(r)$ as above implies a neglect of 3-body correlations. Having established the effectiveness of equation Equation 3.21, we may expect that binning the several nearest neighbor distances in $s(r)$, rather than just the closest, will allow us to gather more statistics while maintaining accuracy. With such a redefinition of $s(x)$, the normalization of P_O^S goes from $\frac{1}{N}$ to $\frac{1}{N(N-1)}$.

3.4.2 Validation and benchmarking

Several researchers in the Glotzer group have independently verified that the method is sufficiently precise for their investigations of various particle shapes and phases. The implementation has been validated to five significant digits of precision

against NpT simulations for hard disks and polygons at only a single state point each due to the computational cost of high precision NpT volume data. This seems sufficient thermodynamic validation, but additional state points and shapes will continue to be investigated.

The analysis code does not noticeably impact the speed of an NVT simulation since it only needs to run at a frequency comparable to the autocorrellation interval. Future work will include evaluation of wall-clock time required to obtain pressure–volume measurements of a given precision in both NpT and NVT. The factors affecting the convergence time of the pressure measurement vary between particle shapes and state points, so several different systems will need to be compared.

By several approaches and concurrence with literature, we justified an expression for volume perturbation approaches to pressure measurement in terms of the probability of introducing overlaps through trial volume moves in a hard particle system. We presented a histogramming method to generate the required probability value, optimized for scalar pressure of convex hard bodies.

To prepare for publication, we will provide validation data obtained by comparison to NpT simulations of solids and fluids in two and three dimensions. We are also preparing benchmark results demonstrating the catastrophic slow down of NpT sampling in large system sizes and the high performance of our implementation.

Future work includes implementing the extensions to measuring pressure for non-convex particles and to measuring the full pressure tensor.

Potential optimizations take two routes, addressing either the histogram construction or the scale factor to collision. The bisection method to determining collision points is only applicable to scalar pressure (isotropic volume changes) and convex

particles. Rather than sample all bins in the histogram for every particle, it seems likely some sort of Monte Carlo sampling of bins and particles should work, and this will be the focus of future investigation. We know of few shapes (spheres, ellipsoids [41]) for which it is possible to determine analytically the scale factor leading to particle collision. However, for such systems, the bisection method or any other repeated sampling would be unnecessary to construct the histogram and evaluating the probability for overlaps would be $\mathcal{O}(1)$ per particle.

CHAPTER IV

Towards analytic equations of state

This chapter provides a literature review and background information beyond the scope of the journal article for which chapter V is being prepared.

During past work in the Glotzer group, a discussion with Rolfe Petschek led to an *ansatz* (4.7) for an equation of state in the high-pressure limit for systems of hard shapes. It prompted much discussion of the role of particle degrees of freedom in the state functions. What follows are preliminary observations to guide a study of the solid phase state functions of hard particles, which has not yet been undertaken.

We will use the convention of a "compressibility factor" defined as the ratio between the volume of a system at a given pressure, temperature, and quantity, with that of an ideal gas under the same conditions.

$$Z \equiv \frac{V}{V_{\text{ideal}}} = \frac{\beta p}{\rho} \tag{4.1}$$

by the ideal gas law, for thermodynamic $\beta \equiv \frac{1}{k_{\text{B}}}$ and number density $\rho \equiv \frac{N}{V}$.

4.1 Experimental equations of state and our current ansatz

The van der Waals equation serves as a reasonable model for a first attempt at deriving an equation of state for impenetrable hard spheres that are otherwise non-

interacting. If we assume no particle interactions other than excluded volume v_{ex} per particle, we can evaluate the partition function, $\mathcal{Z}$.

$$\mathcal{Z} = \frac{1}{N!}\left[\left(\frac{2\pi m}{h^2\beta}\right)^{\frac{3}{2}}(V - v_{\text{ex}})\right]^N \tag{4.2}$$

Trivially, then,

$$\bar{p} = \frac{1}{\beta}\frac{\partial \ln \mathcal{Z}}{\partial V} = \frac{Nk_{\text{B}}T}{V - v_{\text{ex}}} \tag{4.3}$$

This is fine if the scaling of the free volume ($V - v_{\text{ex}}$ with changes in system volume V can be known, but note that then

$$\bar{p} = \frac{Nk_{\text{B}}T}{V - v_{\text{ex}}}\frac{\partial(-v_{\text{ex}})}{\partial V}. \tag{4.4}$$

Historically, free volume theorists have frequently applied a *Voronoi* tessellation as a starting point. Free volume and the excluded volume for which each particle is responsible can be calculated. Kirkwood reformulates the free volume expression of Eyring and Hirschfelder [49] as

$$\frac{pV}{RT} - 1 = \frac{1}{(V/V_{\text{min}})^{\frac{1}{3}} - 1}, \tag{4.5}$$

where V_{min} is the close-packed volume of a face-centered lattice. Salsburg and Wood [50] generalize the equation of state to

$$\frac{\beta p}{\rho} = 1 + \frac{1}{\left(\frac{V}{V_{\text{min}}}\right)^{\frac{1}{\nu}} - 1}, \tag{4.6}$$

where ν is the number of spatial dimensions, V is the system volume, and V_{min} is the value of V in the close packed limit of the configuration. Salsburg and Wood [50] present a derivation of the exact free volume without the independent approximation. The resulting equation of state, neglecting terms of $\mathcal{O}(1)$ and less, is

$$\frac{\beta p}{\rho} \approx \frac{\nu}{\frac{V}{V_{\text{min}}} - 1}, \tag{4.7}$$

which is equivalent to Equation 4.6 to $\mathcal{O}(1)$ if $\left(\frac{V}{V_{\min}}\right)^{\frac{1}{\nu}}$ is replaced by a Taylor expansion to two terms near $V = V_{\min}$.

Equation 4.7 is a nicely simple relationship at high densities, but the partition function presented is not readily evaluated analytically to give an absolute entropy or Helmholtz free energy. Stillinger, Salsburg, and Kornegay [51] discuss the challenges presented by consideration of collective motion as per [50], formulating an asymptotic expansion in which successive terms treat increasing sizes of collectively moving clusters. The approach is demonstrably convergent in the analytically solvable 1-dimensional system, but the problem quickly becomes very challenging even in 2-dimensions. The authors were able only to evaluate the correlated integrals up to clusters of size 4 in the high-density limit of hexagonally packed disks where the accessible shells in 2-dimensional configuration space could be considered as hexagons. Five years later, Salsburg, Rudd, and Stillinger [52] summarized and updated their previous work, considering clusters of up to size 5 in general and to 6 in a particular model system for which they could compare their results to particle simulation data.

The technique can be generalized to a system with ν dimensions of configuration space and Salsburg and Wood [50] have shown that

$$\frac{\beta p}{\rho} = 1 + \frac{\nu(1 - N^{-1})}{\frac{v}{v_0} - 1} + \mathcal{O}(1) \tag{4.8}$$

$$\rightarrow \frac{\beta p}{\rho} \approx \frac{\nu}{\frac{V}{V_0} - 1} \tag{4.9}$$

Salsburg and Wood [50] presented an equation of state for hard spheres in the high-pressure limit that is equivalent to the ansatz from Rolfe Petschek.

4.2 From spheres to polyhedra: translational and rotational partition function

The Wood equation is general enough to include rotational degrees of freedom in the configuration space dimensionality ν, but it is not as clear how to describe the free volume analytically, and the hyper-volume in configuration space becomes much more confusing. The polyhedron's position within its free volume has an effect on the accessible orientations and the orientation of a particle has an effect on the size and shape of the free volume.

Fortunately, we can get statistical values for both translational and angular free "volume" from Monte Carlo. What remains to be done is to study these statistics to see whether and how they truly map to our ansatz. The mobility of particles ought to tell us what to expect from the equation of state and vice versa.

At high density, free volume theory is applicable and we observe that the phase behavior can be approximate in a first order by the equation $p(V - V_{\min}) = cNk_BT$, resembling a van der Waals equation with free volume $V_{\min}$. The dimensionality factor c is related to the scaling of the free volume, which is expected to be $c = 6$ for anisotropic particles. We observe significant deviations from this ideal value even at very high densities. We conjecture the deviation is due to the presence of unbalance degrees of freedom (rattlers).

We will probably find it useful to understand the relationship between compressibility factors and equations of state for objects of various dimensionalities. Wu and Sadus [53] provide a starting point and references.

CHAPTER V

State functions for hard polyhedron fluids

This chapter includes material being prepared for submission for publication with coauthors Michael Engel, Andrew Schultz, David Kofke, and Sharon C. Glotzer.

5.1 Introduction

The equation of state of hard spheres is a classical problem of statistical mechanics and has been studied extensively using analytical calculations and computer simulations over the last 50 years. Yet, little is known about the thermodynamics of anisotropic particles despite experimental and theoretical interest.

Synthetic techniques abound for producing a variety of different polyhedral nano crystals [54]. Various nano crystals are observed experimentally to self-assemble with strictly hard particle interactions or with other tunable parameters by Henzie, *et al.*, [55]. High-pressure experiments performed by Bian, *et al.*, [56] explore the effects of the orientational order of cuboctahedral nano crystals with varying degrees of roundness in FCC or BCC superlattices.

Computational studies have recently probed the nature of the assembly and packing behavior of hard shapes. Dense packings have been rigorously characterized for Platonic and Archimedean solids [57, 58] as well as for some continuous families of

shapes [59]. Other work seeks to bridge observed assembly behavior and our understanding of packing [60]. Assembly behavior has been reported for an extensive zoology of polyhedra [61] following more targeted research on tetrahedra [62, 63], triangular bipyramids [64], truncated octahedra, rhombic dodecahedra, hexagonal prisms, cubes, gyrobifastigia, and triangular prisms [65]. The assembly behavior of continuous families of parameterized shapes has also been explored, including the interpolation of cubes to octahedra via superballs [66], and progressively truncated tetrahedra [61]. New thermodynamic techniques are changing the way we study families of shapes [67] and will continue to improve our understanding of shape as a thermodynamic parameter.

In the present work, we focus on the role of shape on the equation of state. As examples of hard polyhedra we select 13 shapes representative of different assembly behaviors. At fluid densities, virial theory is applicable (subsection 5.2.1), and we apply the scaled particle theory results of Song and Mason [68].

We calculate the state functions of 13 polyhedron fluids using NpT hard particle Monte Carlo simulation. Precision is sufficient to distinguish the data from ideal gas behavior at number densities down to 10^{-4}. Polyhedra are selected from different classes of assembly behavior with an attempt to include and expand on shapes common in the literature. Particle asphericity, defined by Equation 5.7, ranges from 1.184 for cubes to 2.853 for obtuse golden rhombohedra. As discussed below, asphericity appears in the second virial coefficient and in many attempts at analytical equations of state for convex hard particles.

We demonstrate the importance of the second virial coefficient in choosing the volume of a hard sphere against which to compare a polyhedron, showing that a

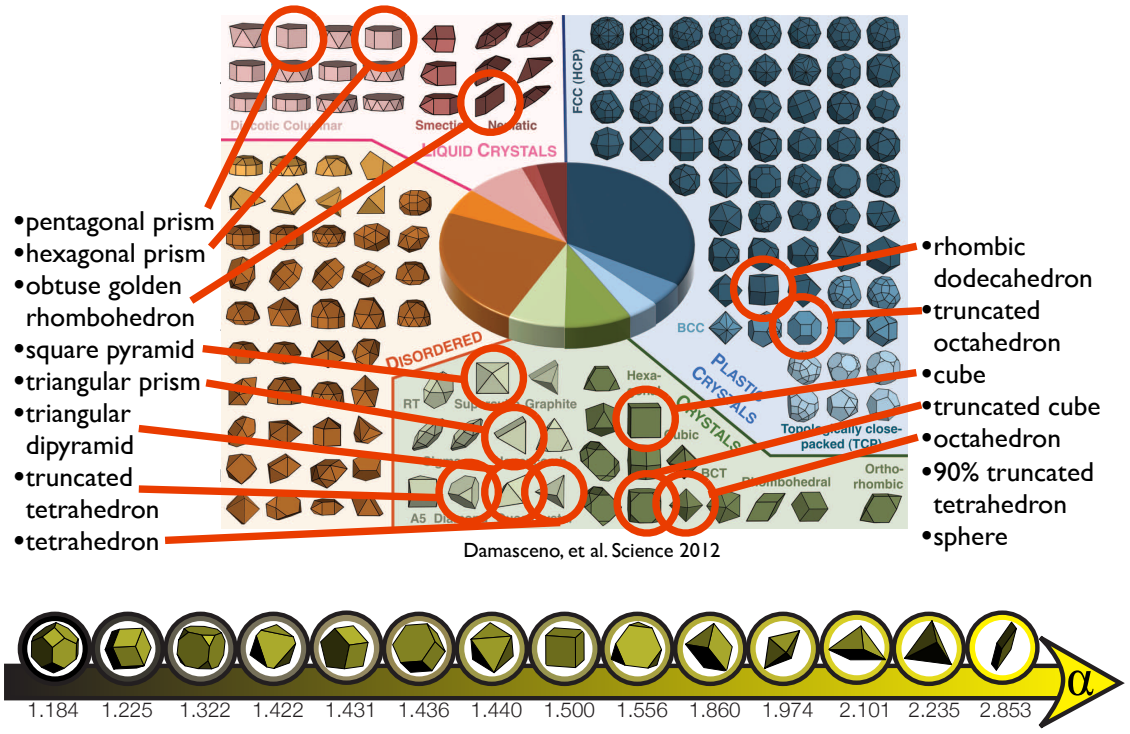


Figure 5.1: Particle shapes were chosen from different classes of assembly behavior with a range of asphericities. The yellow to black color bar shows how data points and trend lines are colored by the asphericity of the represented particle in other figures.

sphere with the resulting effective volume is a good approximation at low density. The compressibility factor, Z, for a polyhedron is always less than that of such an effective hard sphere and asphericity is a good predictor of the rate of deviation.

For more aspherical particles, the Song-Mason equation of state is fairly effective when generalized to polyhedra, improving on earlier Padé approximants appearing in scaled particle theory literature. Asphericity continues to predict the deviation from this equation of state, but the nature of the deviation is not as clearly characterized.

We demonstrate the applicability of a recent general scheme for constructing Padé approximants [69] in these hard polyhedron fluids to high density, though numerical precision in the calculated virial coefficients limits the precision of these equations of state.

5.2 Theoretical background

Equations of state are often represented in terms of the so-called compressibility factor Z, which is essentially a measure of the deviation of a fluid's volume from that of an ideal gas under the same temperature, pressure, and number of particles. Some useful expressions of Z follow.

$$Z \equiv \frac{V}{V_{\text{ideal gas}}} = \frac{pV}{Nk_{\text{B}}T} = \frac{\beta p}{\rho} = \frac{p^*}{\eta} \tag{5.1}$$

where $\rho = N/V$ is the number density, $\eta = Nv_0/V$ is the packing fraction of particles of size v_0, $p^* = \beta p v_0$ is our reduced pressure convention, and thermodynamic $\beta = (k_{\text{B}}T)^{-1}$. Note that Z itself is unitless, though it may be written in terms of non-unitless independent variables.

5.2.1 Virial theory

A virial equation of state takes the form

$$\frac{pV}{Nk_{\mathrm{B}}T} = Z = 1 + \sum_{k=2}^{\infty} B_k(T)\rho^{k-1} \tag{5.2}$$

$$\text{or} \quad \beta p = \rho + \sum_{k=2}^{\infty} B_k(T)\rho^{k} \tag{5.3}$$

where B_k is the k^{th} virial coefficient. Equation 5.2 is unitless and intensive. ρ is intensive, and B_k is intensive with units of $\sigma^{(k-1)\nu}$ for the characteristic length σ in ν dimensions. If we express B_k in terms of characteristic particle volume v_0 then we can write the equation of state:

$$p^* = \beta p v_0 = \eta(1 + \sum_{k=2}^{\infty} \frac{B_k(T)}{v_0^{k-1}}\eta^{k-1}) \tag{5.4}$$

for the packing fraction η where p^* is a unitless reduced pressure. For hard particles, then, we have

$$p^* = \eta + \sum_{k=2}^{\infty} \frac{B_k}{v_0^{k-1}}\eta^{k} \tag{5.5}$$

where $\frac{B_k}{v_0^{(k-1)}}$ is given by a Mayer cluster expansion.

Analytic expressions for some B_k have been determined for some shapes, but B_2 has a simple analytic expression for all convex hard bodies [70] in terms of asphericity α. Asphericity is calculable from the integrated mean curvature r, the surface area s, and the volume v for any characteristic shape function with a clearly defined surface normal.

$$\frac{B_2}{v_0} = 1 + 3\alpha \tag{5.6}$$

$$\alpha = \frac{rs}{3v} \tag{5.7}$$

The integrated mean curvature of a polyhedron, derived as the limiting case of a spheropolyhedron [71], is determined from the edge lengths l_i and the corresponding dihedral angles ϕ_i.

$$r = \sum_i l_i \frac{\phi_i}{12} \tag{5.8}$$

for edge length and dihedral angle l_i and ϕ_i.

5.2.2 Free volume and scaled particle theories

In the first half of the 20th century, free volume theory was used to analyze dense systems of hard spheres. The basic assumptions are that a particle uniquely occupies its own disconnected region of configuration space (single occupancy) in which it moves independently of other particles within its Voronoi cell.

The Birch-Murnaghan equation of state [72, 73] is an attempt to generalize and extend the applicability of the Murnaghan EOS [74, 75]. In a separate evolution, the Rose-Vinet equation of state [76] attempts to correct the excessive stiffness of the Murnaghan equation of state. Meanwhile, Salsburg, Rudd, and Stillinger [51, 52] discuss the considerations of successively larger clusters of dense disks on evaluating free volume.

The Carnahan-Starling equation of state for hard spheres is well-studied and widely used, and lacks only the asphericity term used in the Padé approximants of scaled particle theory. Attempts to generalize departures from spherical particles by incorporating a shape factor use an asphericity term α (Equation 5.7). Nezbeda summarized these results, particularly for spherocylinders, and presented an improved equation of state [77], Equation 5.9. (His more recent work [78] explores alternatives to hard-sphere based approaches when modeling simple fluids with soft potentials.)

$$\frac{\beta p}{\rho} = \frac{1 + (3\alpha - 2)y + (\alpha^2 + \alpha - 1)y^2 - \alpha(5\alpha - 4)y^3}{(1 - y)^3} \tag{5.9}$$

$y = \rho\mathcal{V}$ is the packing ratio

Speedy analyzes the statistical geometry of hard spheres [79], presenting an equation of state (5.10) involving the dimensionality of the system and an interesting geometric construction. The volume uniquely attributable to clusters of sizes zero and up is defined V_j, the volume that lies within one diameter, σ, of j and only j particle centers.

$$\frac{\beta p}{\rho} = 1 + \frac{\sigma}{2\nu}\frac{S_0}{V_0} \tag{5.10}$$

The geometric construction employed avoids the disconnected requirement of free volume in the Salsburg and Wood analysis to give a statistical representation valid at low densities that also detects the liquid-solid phase transition. V_0 sites correspond to cavities, but can be related to sphere sites in a corresponding $N+1$ sphere system. This is the foundation for the work of Sastry, *et al.*, in discussing free volume and pressure of hard spheres in simulations through tesselating cell construction and analysis [48]. Equation Equation 5.10 is modified to consider particle free volumes instead of cavity volumes.

$$\frac{\beta p}{\rho} = 1 + \frac{\sigma}{2\nu}\left\langle\frac{s_f}{v_f}\right\rangle \tag{5.11}$$

The accompanying analysis techniques provide an effective approach to measurements on simulation data, but don't do much to advance theory.

Song and Mason have introduced a now well-known equation of state [68] for hard

Table 5.1: Some expressions for equations of state of varying complexity.

virial:	$\frac{\beta p}{\rho} = 1 + \sum_{k=2}^{\infty} B_k \rho^{k-1}$
Carnahan-Starling:	$\frac{\beta p}{\rho} \approx \frac{1+\eta+\eta^2-\eta^3}{(1-\eta)^3}$
Nezbeda:	$\frac{\beta p}{\rho} \approx \frac{1+(3\alpha-2)\eta+(\alpha^2+\alpha-1)\eta^2-\alpha(5\alpha-4)\eta^3}{(1-\eta)^3}$
Song-Mason:	$\frac{\beta p}{\rho} \approx 1 + \frac{\eta\left((1+3\alpha)-(2+3\alpha-3\alpha^2)\eta+\left(1+\left[\left(\frac{B_4}{v_0^3}\right)_{HS}-12\right]\alpha-7\alpha^2\right)\eta^2\right)}{(1-\eta)^3}$

convex bodies that seems to be oriented to use in isotropically (un)structured fluids.

$$\frac{\beta p}{\rho} \approx 1+\frac{\eta}{(1-\eta)^3}\left((1+3\alpha)-(2+3\alpha-3\alpha^2)\eta+\left(1+\left[\left(\frac{B_4}{v_0^3}\right)_{HS}-12\right]\alpha-7\alpha^2\right)\eta^2\right) \tag{5.12}$$

$\left(\frac{B_4}{v_0^3}\right)_{HS}$ is the fourth virial coefficient for the hard sphere model, which is exactly calculable. In three dimensions, $\left(\frac{B_4}{v_0^3}\right)_{HS} \approx 18.3647684$.

An equation of state for a rigid body of tetrahedrally arranged hard spheres is given by Abascal and Bresme [80]. Potentially important unification of theory originally developed by Kirkwood is presented in a short paper by Khanpour [81] in 2011.

5.3 Methods

5.3.1 Thermodynamic hard particle Monte Carlo

We used NpT hard particle Monte Carlo simulations to measure packing fraction at state points from a reduced pressure of 10^{-4} to the freezing transition in thirteen hard shapes.

Metropolis Monte Carlo volume changes are performed with the acceptance criterion

$$\text{acc}(A \to B) = \min\left[1, \exp\left(-\beta\delta U - \beta p(V_B - V_A) + N \ln\frac{V_B}{V_A}\right)\right].$$

The change in energy δU is either zero or infinity, determined by an overlap check.

The conventions associated with Equation 5.1 make it convenient to operate with state variables p^* and η or with p and ρ, for particles of unit volume, interchangeably. The acceptance criterion is then

$$\mathrm{acc}(A \to B) = \min\left[1, \exp\left(-p^* \frac{(V_B - V_A)}{v_0} + N \ln \frac{V_B}{V_A}\right)\right].$$

5.3.2 Virial coefficients from Mayer sampling Monte Carlo

A recent recipe for the construction of Padé approximants [69] takes an arbitrary number of virial coefficients as inputs. When applied to hard particles, the resulting equation of state can be expressed in an exponential form.

$$Z(\rho) = \exp\left(N_2\bar{\rho} + N_3\bar{\rho}^2 + \cdots + N_J\bar{\rho}^{J-1}\right) \tag{5.13}$$

for reduced density $\bar{\rho} = \rho\sigma^3$, with σ as a characteristic length scale and coefficients N_i calculated as by Barlow *et al.* [69].

The first 8 virial coefficients for all of the polyhedra studied are calculated numerically by our coauthors, Andrew Schultz and David Kofke, who evaluate the Mayer cluster integrals numerically by sampling particle configurations with a technique called Mayer Sampling Monte Carlo (MSMC). MSMC code is implemented on GPUs and discussed in a separate paper [82].

5.3.3 Effective sphere

From Equation 5.6 it is clear that we can match the second virial coefficient of any convex hard shape to that of a sphere of an appropriate size, which we can calculate.

$$\frac{B_2}{v_0} = 1 + 3\alpha \tag{5.14}$$

$$\beta p = \rho + v_0(1 + 3\alpha)\rho^2 + \mathcal{O}(\rho^3) \tag{5.15}$$

$$\frac{v_{\text{es}}}{v_0} = \frac{1}{4} + \frac{3}{4}\alpha \tag{5.16}$$

We calculate α for each of our polyhedra using Equation 5.7 and Equation 5.8. Effective sphere volumes are then given by Equation 5.16.

5.4 Results

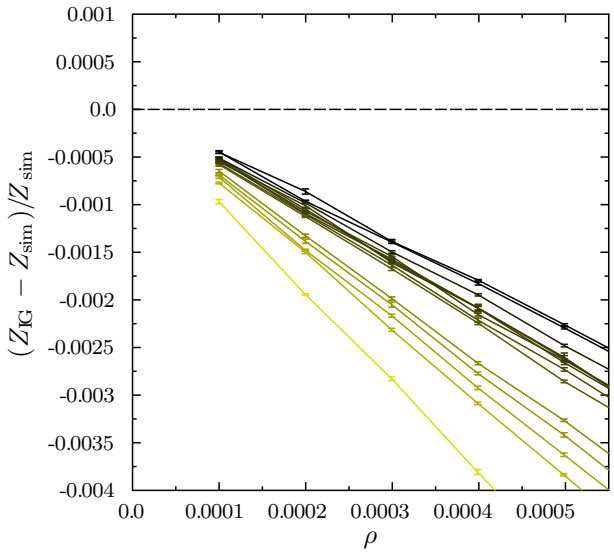


Figure 5.2: At low densities, measurement precision on the order of $\mathcal{O}(\eta)$ is sufficient to distinguish the compressibility factor Z of a finite-sized particle from one (that of an ideal gas). We gathered data down to a packing fraction of $\eta = 10^{-4}$, or number density $\rho = 10^{-4}v_0^{-3}$ for particles of volume v_0.

The aphericity is a parameter in the second virial coefficient as well as in the Nezbeda, Song-Mason, and other scaled particle theory equations of state. We cal-

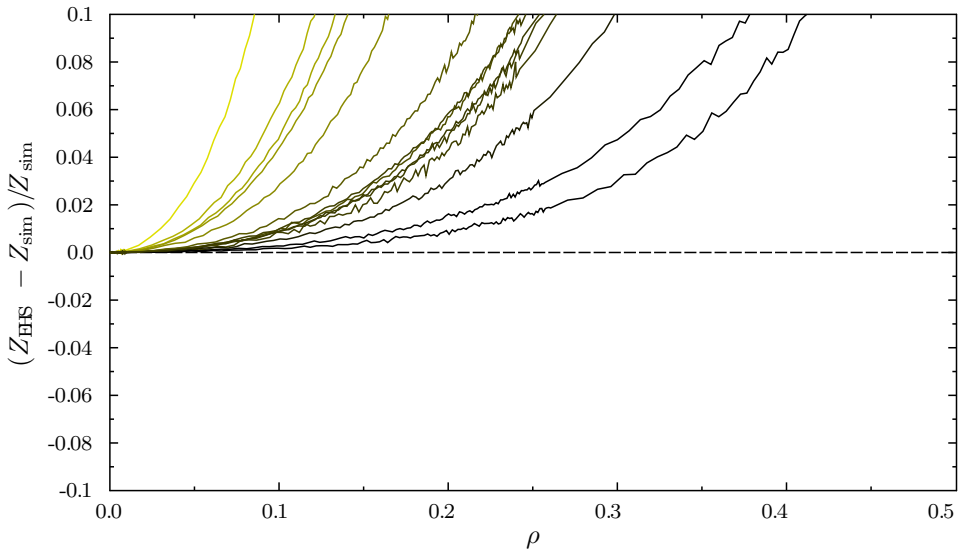


Figure 5.3: For the shapes studied, finding an *effective sphere* by matching second virial coefficients is a quick and effective way to find an upper bound on the compressibility factor and the effective sphere VEOS12 is always a better match to simulation data than VEOS2 (not shown). It is worth noting that all 12 known virial coefficients for hard spheres are positive. B_2 through B_4 for the studied shapes are greater than those for the sphere (all of which are fully analytic) when compared at the same particle volume. The trend is broken by the gyrobifastidium in the fifth virial coefficient, which is less than that of the equal-sized sphere. Several of the more aspherical shapes have $B_6 < 0$.

culated the asphericity α for each polyhedron by Equation 5.7, determining that our polyhedra range from $\alpha = 1.184$ to $\alpha = 2.853$ (Figure 5.1).

It is customary to express compressibility factor in terms of number density ρ. In order to perform illustrative comparisons between state functions, we present the deviations between data in terms of ρ as well, though this means putting our dependent variable on the horizontal axis. Error bars for pressure in terms of density have been estimated from our simulation data, in which density is in terms of pressure.

Deviation from ideal gas behavior can be detected at arbitrarily low density according to the chosen measurement precision (Figure 5.2). We find that our data is of sufficient precision to distinguish the deviation from ideal gas behavior in the sparsest systems studied. At the same time they differentiate themselves from hard spheres of the same volume. To demonstrate the role of particle shape in low density

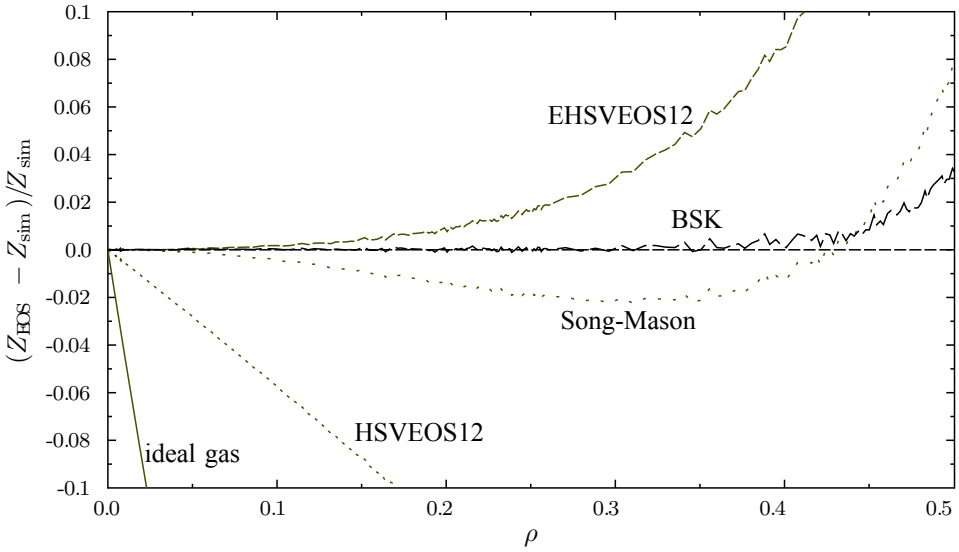


Figure 5.4: We show the suitability of different equations of state to the relatively spherical truncated octahedron. Plots show the relative deviation between the equations of state and the measured data from simulation. Comparisons include $Z = 1$ (ideal gas), the hard sphere 12^{th} order virial equation of state for a sphere of the same particle volume (HSVEOS12), the effective hard sphere virial equation of state (EHSVEOS12) using Equation 5.16, the Song–Mason equation of state from Equation 5.12, and the exponential approximant due to Barlow, Schultz, and Kofke (BSK) from Equation 5.13. Error bars are omitted for clarity. In all but the exponential approximant, error bars are on the order of the visible fluctuations.

systems, it is instructive to compare hard polyhedra to hard spheres of an appropriate effective volume. We recognize that initial departure from ideal gas behavior appears in the second virial coefficient as

$$\frac{B_2}{v_0} = 1 + 3\alpha.$$

By matching the (analytic) second virial coefficients, an effective sphere volume can be found that much better approximates the low density behavior of all particles (Figure 5.3). An anisotropic convex particle has the same B_2 as a sphere of size $v_{\text{es}} = \left(\frac{1}{4} + \frac{3}{4}\alpha\right)$.

Despite the expected differences in the cluster integrals of aspherical particles, an equation of state based on the effective sphere is an improvement over the second order virial equation of state in all cases examined. Results for the most spherical

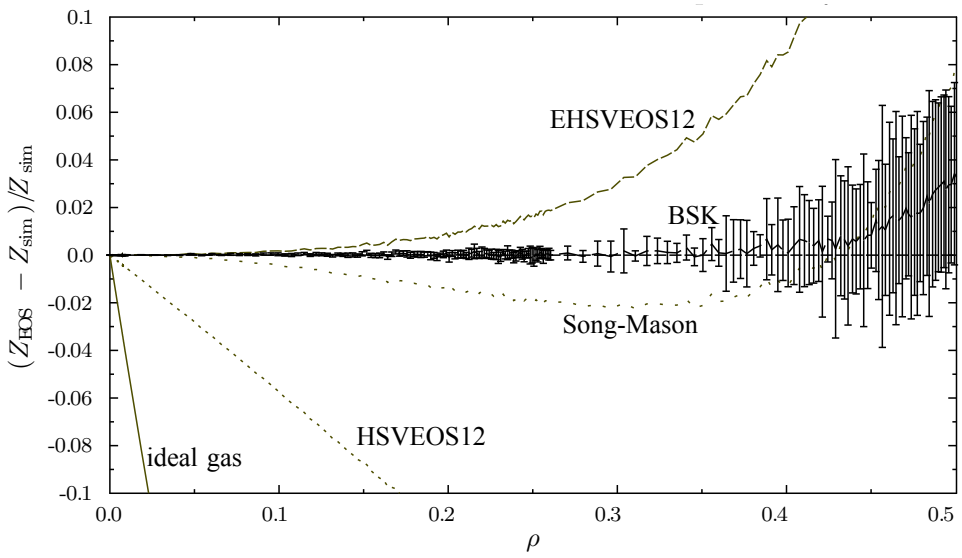


Figure 5.5: For completeness, the error bars for the exponential approximant in Figure 5.4 are shown here. The large error bars represent the accumulated uncertainty due to higher order virial coefficients all calculated with Mayer Sampling Monte Carlo.

shape, the truncated octahedron, are shown in Figure 5.4 and Figure 5.5. Particles with greater asphericity deviate from effective hard sphere behavior more dramatically and at lower densities.

Higher density behavior is better captured by equations of state incorporating α, indicating asphericity is at least somewhat related to higher order cluster integrals than just B_2. For less spherical particles, the Song-Mason equation of state describes particles better than an effective sphere. The approximant created by the recipe of Barlow *et al.*describes even nearly spherical shapes better than the Song-Mason equation of state, within a few percent up to near the freezing transition (Figure 5.6 and Figure 5.7).

5.5 Conclusions

No general analytic or semi-analytic equation of state exists to describe hard polyhedron fluids to high density. However, we have identified several ways to ap-

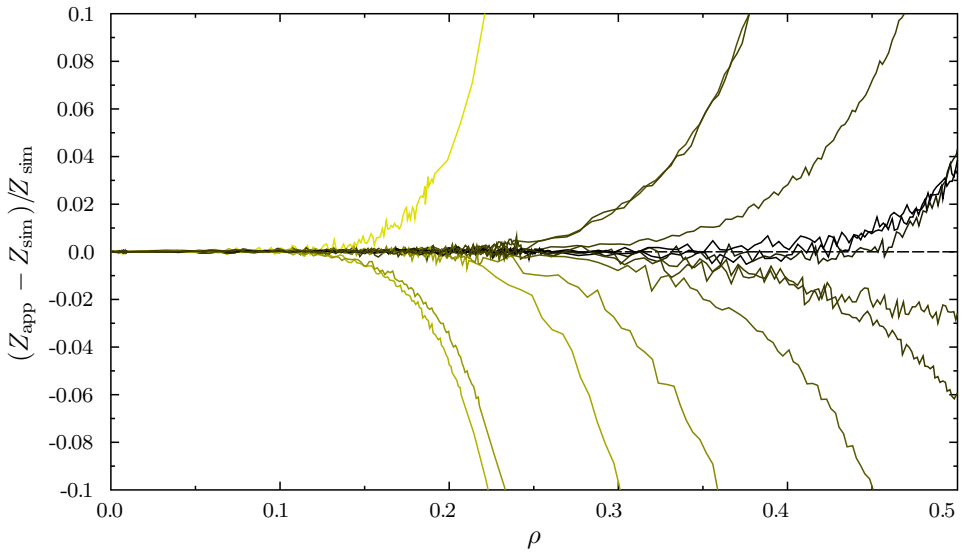


Figure 5.6: The exponential approximant is accurate to within a few percent for all shapes studied up to densities over ten percent. Unlike the effective sphere VEOS12, its deviation at higher density is not entirely predicted by particle asphericity. It would be interesting to evaluate the approximant recipe in other contexts of anisotropic potentials or potentials of different symmetries.

proximate or find the limiting behavior of assembly precursors in solutions of colloids or nanoparticles that may be useful to experimentalists. We have shown that an effective sphere can reproduce the pressure – volume relationship in particles with asphericity on the order of that found in the Platonic solids and other common polyhedra. We show that a recipe developed for producing Padé approximants from virial coefficients is surprisingly applicable to hard particles as well as the soft potentials for which it was designed. From this work we also grow our intuition of the densities at which finite sized objects begin to "notice" each other and the emerging role of shape (not just particle size) as density is increased.

We are thus reminded that when comparing systems of differently shaped particles, it is not sufficient to use a universal length scale based on unit volume or some other arbitrary quantity. A length scale derived from the second virial coefficient is probably a better first guess, but it is evident that comparing the thermodynamic

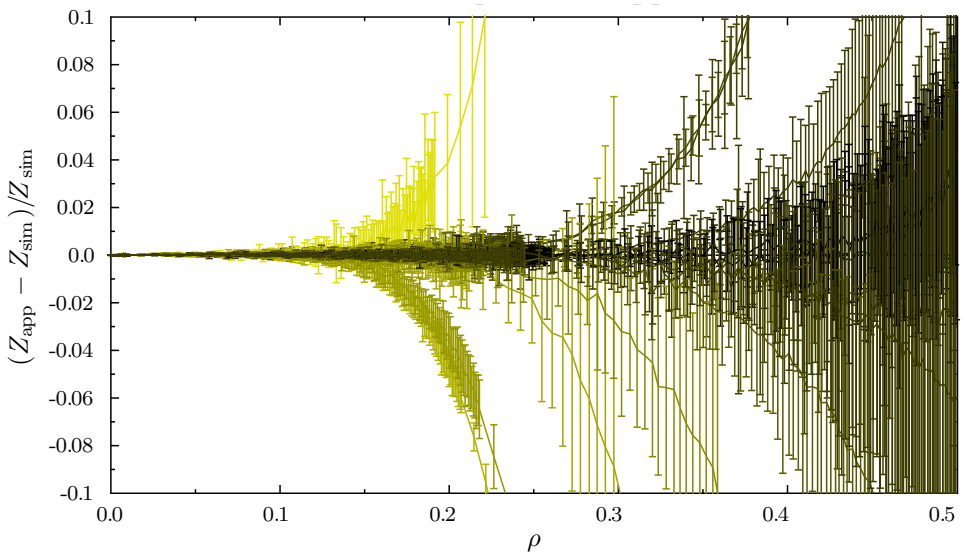


Figure 5.7: Error bars are shown for the data in Figure 5.6

properties of different particles requires care in considering both particle shape *and size*. This author conjectures that in order to derive meaningful information from comparisons of hard shapes, one must devote some consideration to the application. For instance, if one desires to self-assemble a metamaterial with optical properties, the relevant length scale is given by the final density of the material to be active at particular wave lengths. Conveniently, the scaling behavior of hard particle state functions is simple. The compressibility factor, Z, is unaffected by temperature or particle size. That is, a relationship between reduced pressure and packing fraction can be calculated for hard particles of arbitrary size or temperature and applied to particles of the same shape at any size or temperature (Equation 5.1).

CHAPTER VI

Diffraction simulation

This dissertation work included the development of rudimentary diffraction pattern simulation code for application to colloidal and nano particle super structures. This chapter describes basic scattering and diffraction theory, numerical models, and applications to systems of spherical and non spherical particles. It includes a general method for performing the Fourier transform that I have not been able to find in literature. Many software packages exist to model scattering from nano- to mesoscale structures [83–87], but when individual particle form factors are considered, they are treated statistically rather than individually modeled (presumably for performance reasons). I find it difficult to believe that ours is the first description of a completely general polyhedron Fourier transform, though it may be that the method is peculiarly applicable to the data structures common to computational geometry and not of any broader utility. The formulae presented here have been implemented in software for use within the Glotzer group to be integrated with the in-house `freud` analysis software framework. I hope that, with additional work, these tools will ultimately be contributed to open source software for diffraction modeling of mesoscale structures.

It is common for computational scientists to simulate structure factors for their simulated systems through Fourier transforms. A grid of particle density can be processed with a discrete Fast Fourier Transform (FFT) [88], but it is unclear how best to incorporate arbitrary particle orientations and off-lattice positions with minimal discretization artifacts.

We developed code to simulate diffraction patterns of particles with various form factors, scattering densities and arbitrary orientations. Using analytic expressions for particle form factors and explicitly evaluating scattering density at pre-calculated k-space points yielded good results with reasonable computational cost for small systems. Our method is more general than methods for specific shapes or that first decompose an object into simpler shapes [89]. The approach is particularly suited for application to known structures achieved in simulation for comparison to diffraction patterns obtained through microscopy.

6.1 Background

The following discussion of classical scattering of waves largely follows Ibach and Luth [90].

Consider a small illuminated sample being observed at point B. The amplitude of the probe wave (before encountering the sample) is given by $A_0 \exp i(\mathbf{k}_0 \cdot \mathbf{r} - \omega_0 t + \phi)$ for a coherent plane wave probe with wave vector $\mathbf{k}_0$. Choosing to measure phase relative to the coordinate system origin, we choose $\phi = 0$. The incoming wave is described by

$$A_P(\mathbf{r}, t) = A_0 e^{i\mathbf{k}_0 \cdot \mathbf{r} - i\omega_0 t} \tag{6.1}$$

for a scatterer at point P at location $\mathbf{r}$ in the sample.

If we assume that spherical waves are elastically scattered from isotropic (but possibly complex-valued) scattering density $\rho(\mathbf{r})$ at fixed P,

$$A_B(\mathbf{R},\mathbf{r},t) = \frac{1}{|\mathbf{R}-\mathbf{r}|} A_P(\mathbf{r},t)\rho(\mathbf{r}) \exp\left(i\mathbf{k}_{1,\mathbf{r},B}\cdot(\mathbf{R}-\mathbf{r})\right) \tag{6.2}$$

observed at point B at location $\mathbf{R}$. For B in the far field, we can make some approximations based on large $\mathbf{R}$.

$$\lim \frac{1}{|\mathbf{R}-\mathbf{r}|} \simeq \frac{1}{R} \tag{6.3a}$$

$$\mathbf{k}_{1,\mathbf{R},B} \simeq \mathbf{k}_1 \tag{6.3b}$$

$$\mathbf{k}_1 \cdot \mathbf{R} \simeq kR \tag{6.3c}$$

That is to say that all scattered waves observed at B are parallel to $\mathbf{R}$ and are attenuated by the same amount.

$$A_B(\mathbf{R},\mathbf{r},t) = \frac{A_P(\mathbf{r},t)}{R}\rho(\mathbf{r})e^{i\mathbf{k}_1\cdot(\mathbf{R}-\mathbf{r})} \tag{6.4}$$

$$= \frac{A_0}{R}e^{i(\mathbf{k}_1\cdot\mathbf{R})}e^{-i\omega_0 t}\rho(\mathbf{r})e^{i\mathbf{k}_0\cdot\mathbf{r}}e^{-i\mathbf{k}_1\cdot\mathbf{r}} \tag{6.5}$$

$$= \frac{A_0}{R}e^{i(kR)}e^{-i\omega_0 t}\rho(\mathbf{r})e^{-i(\mathbf{k}_1-\mathbf{k}_0)\cdot\mathbf{r}} \tag{6.6}$$

$$\propto e^{-i\omega_0 t}\rho(\mathbf{r})e^{-i(\mathbf{k}_1-\mathbf{k}_0)\cdot\mathbf{r}} \tag{6.7}$$

Scattering intensity is determined as $|A|^2 = A^*A$, meaning the exponential term in time will go to unity. We will ignore it for clarity from now on. Then

$$A_B(\mathbf{R},\mathbf{r}) \propto \rho(\mathbf{r})e^{-i\mathbf{K}\cdot\mathbf{r}}\ , \quad \text{for } \mathbf{K} \equiv \mathbf{k}_1 - \mathbf{k}_0. \tag{6.8}$$

The amplitude at B due to all scattering centers is found by integrating $\mathbf{r}$ over the sample.

$$A_B(\mathbf{R}) \propto \int \rho(\mathbf{r})e^{-i\mathbf{K}\cdot\mathbf{r}}d\mathbf{r} \tag{6.9}$$

6.2 Calculation of scattering intensity

The measured scattering intensity, proportional to the square of the amplitude due to all scattering locations P, is then

$$I(\mathbf{K}) \propto \left| \int \rho(\mathbf{r}) e^{-i\mathbf{K}\cdot\mathbf{r}} d\mathbf{r} \right|^2 = |P(\mathbf{K})|^2 \ , \text{ where } P(\mathbf{K}) = \mathcal{F}(\rho). \tag{6.10}$$

Equation (6.10) shows that scattering intensity is the square of the Fourier transform of the scattering density.

6.2.1 Separability of form factor and structure factor

A given scatterer α has a scattering density $\rho_\alpha(\mathbf{r}')$ in its own coordinate system. Translating that particle to a location $\mathbf{r}_\alpha$ is equivalent to convolving with a delta function. We will use $*$ to denote the convolution operator. Let $\delta_\alpha(\mathbf{r}) = \delta(\mathbf{r} - \mathbf{r}_\alpha)$. Then

$$\begin{aligned}
(\delta_\alpha * \rho_\alpha)(\mathbf{r}) &= \int \delta_\alpha(\mathbf{r} - \mathbf{r}')\rho_\alpha(\mathbf{r}')d\mathbf{r}' && (6.11)\\
&= \int \delta(\mathbf{r} - \mathbf{r}' - \mathbf{r}_\alpha)\rho_\alpha(\mathbf{r}')d\mathbf{r}' && (6.12)\\
&= \rho_\alpha(\mathbf{r} - \mathbf{r}_\alpha) && (6.13)\\
\rho(\mathbf{r}) = \rho_\alpha(\mathbf{r} - \mathbf{r}_\alpha) &= (\delta_\alpha * \rho_\alpha)(\mathbf{r}) && (6.14)
\end{aligned}$$

In general, the distribution ρ_L can describe many particles.

$$\rho_L(\mathbf{r}) = \sum_\alpha \delta(\mathbf{r} - \mathbf{r}_\alpha) \tag{6.15}$$

$$P(\mathbf{K}) = \int d\mathbf{r} \sum_\alpha \left(\delta(\mathbf{r} - \mathbf{r}_\alpha) * \rho_\alpha(\mathbf{r})\right) e^{-i\mathbf{K}\cdot\mathbf{r}} \tag{6.16}$$

$$= \sum_\alpha \mathcal{F}(\delta_\alpha * \rho_\alpha) \tag{6.17}$$

$$= \sum_\alpha e^{-i\mathbf{K}\cdot\mathbf{r}_\alpha} \mathcal{F}(\rho_\alpha) \tag{6.18}$$

$$= \sum_\alpha F_\alpha(\mathbf{K}) e^{-i\mathbf{K}\cdot\mathbf{r}_\alpha} \tag{6.19}$$

where F is the scattering form factor of particle α. For many particles with the same scattering profile ρ_p, $\rho(\mathbf{r})$ takes a similar form to (6.14).

$$\rho(\mathbf{r}) = \sum_\alpha \left(\delta(\mathbf{r} - \mathbf{r}_\alpha) * \rho_\alpha(\mathbf{r})\right) \tag{6.20}$$

$$= \left(\sum_\alpha \delta(\mathbf{r} - \mathbf{r}_\alpha)\right) * \rho_p(\mathbf{r}) \tag{6.21}$$

$$\rho(\mathbf{r}) = (\rho_L * \rho_p)(\mathbf{r}) \tag{6.22}$$

$$P(\mathbf{K}) = \mathcal{F}(\rho_L * \rho_p) = \mathcal{F}(\rho_L)\mathcal{F}(\rho_p) = S(\mathbf{K})F(\mathbf{K}) \tag{6.23}$$

Combining (6.23) and (6.10) shows that the scattering intensity is separable into a structure factor S and a form factor F.

$$I(\mathbf{K}) \propto |\mathcal{F}(\rho)|^2 = |\mathcal{F}(\rho_L)|^2 \, |\mathcal{F}(\rho_p)|^2 \tag{6.24}$$

$$I(\mathbf{K}) = |S(\mathbf{K})F(\mathbf{K})|^2 = S^2(\mathbf{K})F^2(\mathbf{K}) \tag{6.25}$$

In reciprocal space, the structure factor contains all of the information about the distribution of scatterers, while the form factor describes the scattering profile. Experimental diffraction measurements have access only to the intensity I, in which

the material structure and particle scattering profile are muddled, and in which phase information is lost due to the squaring.

However in some cases the form factor can be measured, such as by scattering off of a diffuse, randomly arranged sample of particles, and the squared structure factor can be determined by dividing an ordered sample's scattering intensity by that of a disordered sample.

It is convenient to calculate the structure factor and form factor separately (see below).

6.2.2 Structure factor

As we can see from (6.15) and (6.23), the structure factor is the Fourier transform of a set of delta peaks. If we want to simulate a diffraction pattern or perform some sort of periodicity analysis, we must evaluate the structure factor over a region of k-space with each particle in the system contributing an exponential (or trigonometric) function at each point. This leads to consideration of performance versus accuracy considerations.

Explicit FT at reciprocal lattice points

In the special case that we know the lattice parameters *a priori* we may calculate the reciprocal lattice vectors in order to compute a scattering intensity at specific points of interest. For simplicity, first consider one spatial dimension. Recognize that for a periodic structure, $\rho(x) = \rho(x + na)$, leading us to rewrite ρ as a Fourier

series.

$$\rho(x) = \sum_{n=-\infty}^{\infty} \rho_n e^{in\frac{2\pi x}{a}} \tag{6.26}$$

$$\rho_n = \frac{1}{2\pi}\int_0^a dx\, \rho(x) e^{-in\frac{2\pi x}{a}} \tag{6.27}$$

In three dimensions, we write

$$\mathbf{r} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3 \tag{6.28}$$

$$\mathbf{G}\cdot\mathbf{r} = 2\pi m,\ m \in \text{ integers} \tag{6.29}$$

$$\rho(\mathbf{r}) = \sum_{\mathbf{G}} \rho_{\mathbf{G}} e^{i\mathbf{G}\cdot\mathbf{r}} \tag{6.30}$$

Note that, for $\mathbf{G} = (h\mathbf{g}_1, k\mathbf{g}_2, l\mathbf{g}_3)$, equation (6.29) is satisfied for $\mathbf{g}_i \cdot \mathbf{a}_j = 2\pi\delta_{ij}$, or

$$\mathbf{g}_1 = 2\pi\frac{\mathbf{a}_2\times\mathbf{a}_3}{\mathbf{a}_1\cdot(\mathbf{a}_2\times\mathbf{a}_3)} \text{ and cyclic permutations.} \tag{6.31}$$

Using the Fourier series representation of $\rho(\mathbf{r})$, we see that

$$I(\mathbf{K}) \propto \frac{|A_0|^2}{R^2}\left|\sum_{\mathbf{G}}\rho_{\mathbf{G}}\int d\mathbf{r}\, e^{i\mathbf{G}\cdot\mathbf{r}} e^{-i\mathbf{K}\cdot\mathbf{r}}\right|^2. \tag{6.32}$$

The integral is a familiar Fourier transform and evaluates to $\prod_\alpha \delta(K_\alpha - G_\alpha)$ giving us the Laue condition $\mathbf{K} = \mathbf{G}$. The structure factor, then, is only non-zero (and only needs to be evaluated) at reciprocal lattice sites. If we do not know a unit cell, but we have a simulation box with periodic boundary conditions, we effectively have a single large cell for which we can calculate and evaluate $\mathbf{K}$ values. We must be careful, though to ignore k-space features of length $\frac{2\pi}{a_\alpha}$ in the $\mathbf{g}_\alpha$ direction, which result from the periodicity of the box.

The Fourier coefficients are given by

$$\rho_{\mathbf{G}} = \rho_{hkl} = \frac{1}{V_{\text{cell}}}\int_{\text{cell}} \rho(\mathbf{r}) e^{-i\mathbf{G}\cdot\mathbf{r}} d\mathbf{r} \tag{6.33}$$

$$\rightarrow I(\mathbf{K}) \propto \frac{|A_0|^2}{R^2}\left|\frac{1}{V_{\text{cell}}}\int_{\text{cell}} \rho(\mathbf{r}) e^{-i\mathbf{K}\cdot\mathbf{r}} d\mathbf{r}\right|^2,\ \mathbf{K} = \mathbf{G}. \tag{6.34}$$

Then, for scatterers α in a unit cell with local coordinates $\mathbf{r}_\alpha = u_\alpha \mathbf{a}_1 + v_\alpha \mathbf{a}_2 + w_\alpha \mathbf{a}_3$,

$$I(\mathbf{K}) = I(h,k,l) \propto |P_{hkl}|^2 \,,\; P_{hkl} = \sum_\alpha f_\alpha S_{\alpha,hkl} \tag{6.35}$$

$$S_{\alpha,hkl} = e^{-i\mathbf{G}\cdot\mathbf{r}_\alpha} = e^{-i2\pi(hu_\alpha + kv_\alpha + lw_\alpha)} \tag{6.36}$$

For particles with identical form factors, such that $\sum f_\alpha S_\alpha = f \sum S_\alpha$, we may write

$$S_{hkl} = \sum_\alpha S_{\alpha,hkl} = \sum_\alpha e^{-i2\pi(hu_\alpha + kv_\alpha + lw_\alpha)} \tag{6.37}$$

In this way we may simplify calculation of a diffraction pattern, but we also gain insight into the contributions of each particle in the unit cell to the scattering intensity at the reciprocal lattice points. Note that systems of non-identical particles may still contain a small number of particles for which contributions to scattered wave amplitude can be calculated.

Symmetry

We may further reduce the number of k-space points to evaluate through symmetry. For elastic scattering, $\rho(\mathbf{r})$ is real valued [90]. Then

$$\rho_{hkl} = \rho^*_{\bar{h}\bar{k}\bar{l}} \rightarrow I_{hkl} = I_{\bar{h}\bar{k}\bar{l}} \text{ (Friedel's rule)} \tag{6.38}$$

6.3 Calculation of form factor

The following integrals tend to lead to Bessel functions. The 0^{th} order spherical Bessel function of the first kind, $j_0(x)$, is the unnormalized sine cardinal function and will be abbreviated in the following equations according to the standard mathematical convention $\text{sinc}(x) = \frac{\sin(x)}{x}$ for $x \neq 0$ and $\text{sinc}(0) = 1$, the limiting value.

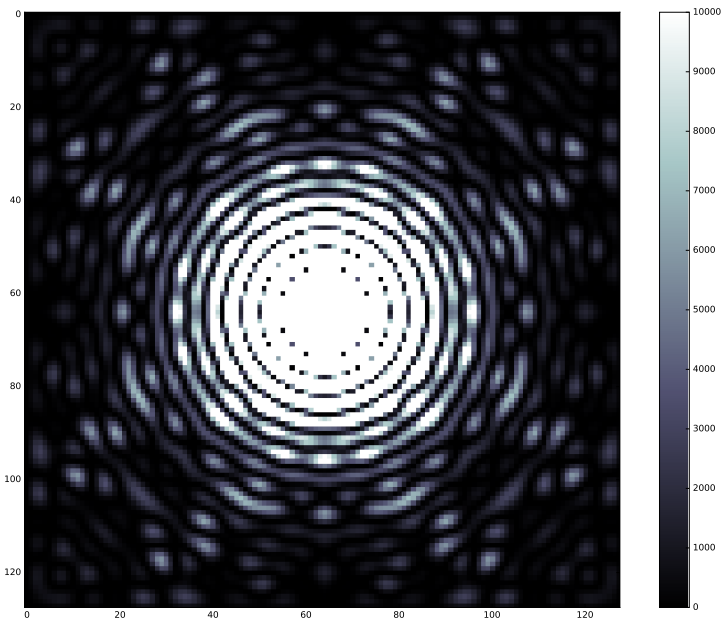


Figure 6.1: Fast Fourier Transform (discrete) of a sphere of uniform density.

6.3.1 Uniform sphere

We define a uniform sphere of radius R centered at the origin of its coordinate system to have a density

$$\rho(\mathbf{r}) = \begin{cases} \rho_0 & r <= R \\ 0 & r > R \end{cases} \tag{6.39}$$

We then seek to evaluate the Fourier transform

$$F(\mathbf{K}) = \int d\mathbf{r}\, \rho(\mathbf{r}) e^{-i\mathbf{K}\cdot\mathbf{r}}. \tag{6.40}$$

An analytic solution allows evaluation at any K value without interpolation or transformation of pre calculated results. The integrand of the form factor has support defined by the sphere and the integral can be rewritten.

$$F(\mathbf{K}) = \int d\mathbf{r}\, \rho(\mathbf{r}) e^{-i\mathbf{K}\cdot\mathbf{r}} \tag{6.41}$$

$$= \int_0^{2\pi} d\phi \int_0^{\pi} d\theta \int_0^{R} dr\, \rho_0 r^2 \sin\theta e^{-iKr\cos\theta} \tag{6.42}$$

Let $\mu = -\cos\theta$. $d\mu = \sin\theta\, d\theta$.

$$F(\mathbf{K}) = 2\pi\rho_0 \int_{-1}^{1} d\mu \int_0^{R} dr\, r^2 e^{iKr\mu} \tag{6.43}$$

$$= 2\pi\rho_0 \int_0^{R} dr\, r^2 \left[\frac{1}{iKr} e^{iKr\mu}\right]_{-1}^{1} \tag{6.44}$$

$$= 2\pi\rho_0 \int_0^{R} dr\, r^2 \frac{e^{iKr} - e^{-iKr}}{iKr} \tag{6.45}$$

$$= \rho_0 \int_0^{R} dr\, 4\pi r^2 \operatorname{sinc} Kr \tag{6.46}$$

$$= \rho_0 4\pi \left(\frac{\sin KR}{K^3} - \frac{KR}{K^3}\cos KR\right) \tag{6.47}$$

$$F(\mathbf{K}) = \rho_0 \frac{4\pi R}{K^2} \left(\operatorname{sinc}(KR) - \cos(KR)\right) \tag{6.48}$$

The integral in (6.48) can be evaluated by considering the recursion relations of spherical Bessel functions, but I just used Mathematica. As a check, we can confirm that the Fourier transform for unity density at $K = 0$ gives the volume of the sphere.

$$\lim_{K\to 0} \frac{4\pi R}{K^2} \left(\operatorname{sinc}(KR) - \cos(KR)\right) = \frac{4\pi}{3} R^3 \tag{6.49}$$

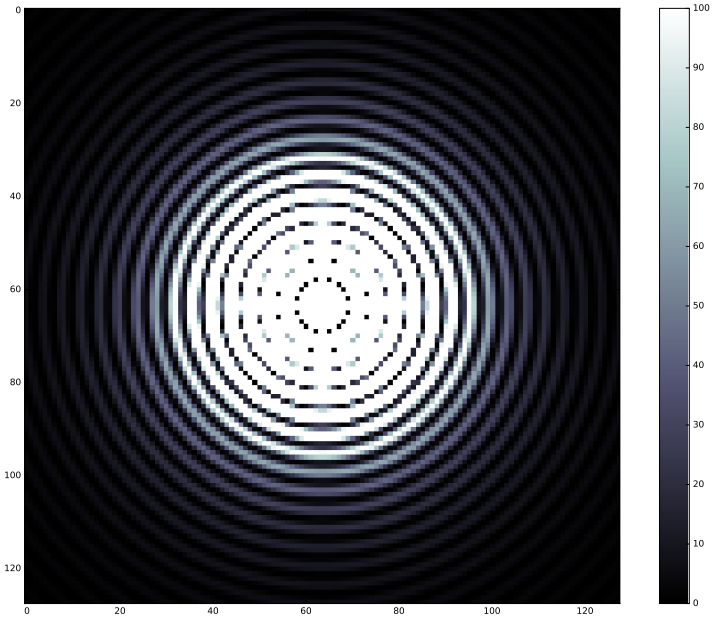


Figure 6.2: Analytic solution to sphere Fourier transform shows that the discrete FT (Figure 6.1) includes many artifacts that could confuse simulated diffraction analysis.

6.3.2 Polygons and polyhedra

We start by considering the Fourier transform of a polygonal surface Σ, defined by a shape function, the distribution $\rho(\mathbf{r})$. In three dimensions, the Fourier transform is given by

$$\mathcal{F} = \int_{\Re^3} \rho(\mathbf{r}) e^{-i\mathbf{k}\cdot\mathbf{r}} d^3\mathbf{r} \tag{6.50}$$

but the function only has polygonal support in the plane of the surface.

$$\mathcal{F} = \exp(-i\mathbf{k}\cdot\hat{\mathbf{n}}d) \int_{\Sigma} \rho(\mathbf{s}, \mathbf{d}\hat{\mathbf{n}}) e^{-i(\mathbf{k}-(\mathbf{k}\cdot\hat{\mathbf{n}})\hat{\mathbf{n}})\cdot\mathbf{s}} d^2\mathbf{s} \tag{6.51}$$

where $\mathbf{s}$ is the projection of $\mathbf{r}$ to the polygonal surface, $\hat{\mathbf{n}}d$ is is the projection of $\mathbf{r}$ along the surface normal, and d the distance of the surface from the origin. We then consider just the Fourier transform of a polygon in two dimensions.

McInturff and Simon [91] present a solution in which we use Stokes' theorem to convert the surface integral to a line integral around the edges of the polygon. The derivation (somewhat involved) relies on proper selection of a field $\mathbf{F}$ allowing us to write

$$S(\mathbf{k}) = \int_{\Sigma} \rho(\mathbf{r}) \exp(-i\mathbf{k}\cdot\mathbf{r}) d\mathbf{r} = \int_{\Sigma} \nabla \times \mathbf{F} \cdot d\mathbf{A} \tag{6.52}$$

where $d\mathbf{A} = \hat{\mathbf{n}}\, dx\, dy$ for x, y in the plane of Σ and $\hat{\mathbf{n}} = \hat{\mathbf{z}}$ the plane normal. For $\rho(\mathbf{r}) = \rho_0$ constant over Σ, a suitable choice of F is

$$\mathbf{F} = -i\rho_0 \exp(-i\mathbf{k}\cdot\mathbf{r}) \frac{\mathbf{k}\times\hat{\mathbf{n}}}{|\mathbf{k}|^2}. \tag{6.53}$$

Note that due to the difference in convention adopted for the form of the Fourier transform, the sign on our choice of F is opposite that in [91].

We check that (6.53) satisfies (6.52) as follows. For clarity, assume $\rho_0 = 1$.

Let $\mathbf{r} = x\hat{\mathbf{x}} + y\hat{\mathbf{y}},\ d\mathbf{A} = \hat{\mathbf{z}}\,dx\,dy$.

$$\begin{aligned}\mathbf{F} &= -i\exp(-i\mathbf{k}\cdot\mathbf{r})\frac{\mathbf{k}\times\hat{\mathbf{n}}}{|\mathbf{k}|^2} \\ &= \frac{-i}{|\mathbf{k}|^2}e^{-i(k_x x+k_y y)}(k_y\hat{\mathbf{x}} - k_x\hat{\mathbf{y}}) \end{aligned} \tag{6.54}$$

Note that F_z is zero and the partial derivatives with respect to z are zero.

$$\nabla\times\mathbf{F} = (\partial_x F_y - \partial_y F_x)\hat{\mathbf{z}} \tag{6.55}$$

$$= (-ik_x F_y + ik_y F_x)\hat{\mathbf{z}} \tag{6.56}$$

Note that $\mathbf{k}\times\mathbf{F} = (k_x F_y - k_y F_x)\hat{\mathbf{z}}$.

$$\nabla\times\mathbf{F} = -i\mathbf{k}\times\mathbf{F} = \frac{i^2}{|\mathbf{k}|^2}\exp(-i\mathbf{k}\cdot\mathbf{r})\mathbf{k}\times(\mathbf{k}\times\hat{\mathbf{z}}) \tag{6.57}$$

$$= \exp(-i\mathbf{k}\cdot\mathbf{r})\hat{\mathbf{z}}\frac{\mathbf{k}\cdot\mathbf{k}}{|\mathbf{k}|^2}\ \text{(by triple vector product rule)} \tag{6.58}$$

$$= \exp(-i\mathbf{k}\cdot\mathbf{r})\hat{\mathbf{z}} \tag{6.59}$$

$$S(\mathbf{k}) = \int_\Sigma \exp(-i\mathbf{k}\cdot\mathbf{r})d\mathbf{r} = \int_\Sigma \nabla\times\mathbf{F}\cdot d\mathbf{A} \tag{6.60}$$

Now apply Stokes' theorem.

$$S(\mathbf{k}) = \oint_{\partial\Sigma}\mathbf{F}\cdot d\mathbf{r} = \sum_{n=0}^{N-1}\int_{\mathbf{r}_n}^{\mathbf{r}_{n+1}}\mathbf{F}\cdot d\mathbf{r} \tag{6.61}$$

where we now integrate over each edge of the polygon. To present the integral in a more general form, we would like to parameterize $\mathbf{r}$ in terms of t, $\mathbf{r}_n$, and $\mathbf{r}_{n+1}$. Find $\mathbf{r}(t)$ such that

$$\int_{\mathbf{r}_n}^{\mathbf{r}_{n+1}} d\mathbf{r} = \int_{t_0}^{t_f} p(t,\mathbf{r}_n,\mathbf{r}_{n+1})dt = \mathbf{r}_{n+1} - \mathbf{r}_n = \boldsymbol{l}_n \tag{6.62}$$

where $\boldsymbol{l}_n$ is the n^{th} edge of the polyhedron. To simplify a later integral, we will choose $t_f = 1,\ t_0 = -1$.

$$\mathbf{r}(t) = \frac{t_f - t}{t_f - t_0}\mathbf{r}_n + \frac{t - t_0}{t_f - t_0}\mathbf{r}_{n+1} \tag{6.63}$$

$$= \tfrac{1}{2}\left[(1-t)\mathbf{r}_n + (1+t)\mathbf{r}_{n+1}\right] \tag{6.64}$$

$$d\mathbf{r} = \tfrac{1}{2}(\mathbf{r}_{n+1} - \mathbf{r}_n)dt = \tfrac{1}{2}\boldsymbol{l}_n dt \tag{6.65}$$

Also note that

$$\tfrac{1}{2}\left[(1-t)\mathbf{r}_n + (1+t)\mathbf{r}_{n+1}\right] = \frac{\boldsymbol{l}t}{2} + \frac{\mathbf{r}_{n+1} + \mathbf{r}_n}{2} = \tfrac{1}{2}\boldsymbol{l}t + \mathbf{c}_n \tag{6.66}$$

where $\mathbf{c}_n$ is the center of the $\mathbf{n}^{\text{th}}$ edge.

$$S(\mathbf{k}) = \sum_{n=0}^{N-1}\int_{\mathbf{r}_n}^{\mathbf{r}_{n+1}} \mathbf{F}\cdot d\mathbf{r} = \frac{1}{2}\sum_{n=0}^{N-1}\int_{-1}^{1}\mathbf{F}\cdot\boldsymbol{l}_n dt \tag{6.67}$$

$$= \frac{-i}{2|\mathbf{k}|^2}\sum_{n=0}^{N-1}\int_{-1}^{1}\exp(-i\mathbf{k}\cdot(\tfrac{1}{2}\boldsymbol{l}t + \mathbf{c}_n))(\mathbf{k}\times\hat{\mathbf{z}})\cdot\boldsymbol{l}_n dt \tag{6.68}$$

$$= \frac{-i}{2|\mathbf{k}|^2}\sum_{n=0}^{N-1}(\mathbf{k}\times\hat{\mathbf{z}})\cdot\boldsymbol{l}_n\, e^{-i\mathbf{k}\cdot\mathbf{c}_n}\int_{-1}^{1} e^{-i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}t}dt \tag{6.69}$$

$$= \frac{-i}{|\mathbf{k}|^2}\sum_{n=0}^{N-1}\hat{\mathbf{z}}\cdot(\boldsymbol{l}_n\times\mathbf{k})\, e^{-i\mathbf{k}\cdot\mathbf{c}_n}\frac{1}{2}\left[\frac{-1}{i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}}(e^{-i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}} - e^{i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}})\right] \tag{6.70}$$

$$= \frac{-i}{|\mathbf{k}|^2}\sum_{n=0}^{N-1}\hat{\mathbf{z}}\cdot(\boldsymbol{l}_n\times\mathbf{k})\, e^{-i\mathbf{k}\cdot\mathbf{c}_n}\frac{1}{\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}}\left[\frac{e^{i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}} - e^{-i\frac{\mathbf{k}\cdot\boldsymbol{l}}{2}}}{2i}\right] \tag{6.71}$$

$$S(\mathbf{k}) = \frac{-i}{|\mathbf{k}|^2}\sum_{n=0}^{N-1}\hat{\mathbf{z}}\cdot(\boldsymbol{l}_n\times\mathbf{k})\, e^{-i\mathbf{k}\cdot\mathbf{c}_n}\operatorname{sinc}\tfrac{\mathbf{k}\cdot\boldsymbol{l}}{2} \tag{6.72}$$

The solution is not valid for $|\mathbf{k}| = 0$. McInturff and Simon choose to make F a piecewise function of $\mathbf{k}$ with a different definition for $|\mathbf{k}|^2 = 0$, but it is sufficient to observe that for $|\mathbf{k}|^2 = 0$, the exponential is unity and the Fourier transform is simply $\rho_0 A$, where A is the area of the polygon.

With this derivation as inspiration, we may pursue the Fourier transform of the polyhedron by recognizing that the divergence theorem allows us to turn a volume integral into a surface integral. Again, we will need a suitable choice for a vector field $\mathbf{F}$ but we gain inspiration from McInturff and Simon. Let a polyhedron be defined by a shape function of uniform density over the region Ω.

$$\rho(\mathbf{r}) = \begin{cases} \rho_0 & \mathbf{r} \in \Omega \\ 0 & \text{otherwise} \end{cases} \tag{6.73}$$

$$\mathcal{F} = \int_{\Re^3} \rho(\mathbf{r}) e^{-i\mathbf{k}\cdot\mathbf{r}} d^3\mathbf{r} = \int_{\Omega} \rho_0 e^{-i\mathbf{k}\cdot\mathbf{r}} dV \tag{6.74}$$

$$\frac{S(\mathbf{k})}{\rho_0} = \int_{\Omega} e^{-i\mathbf{k}\cdot\mathbf{r}} dV \tag{6.75}$$

Again, we'll neglect the constant density for the purpose of clarity and try to find $\mathbf{F}$ such that

$$\int_{\Omega} e^{-i\mathbf{k}\cdot\mathbf{r}} dV = \int_{\Omega} \nabla \cdot \mathbf{F}\, dV. \tag{6.76}$$

Try

$$\mathbf{F} = i e^{-i\mathbf{k}\cdot\mathbf{r}} \frac{\mathbf{k}}{|\mathbf{k}|^2}. \tag{6.77}$$

Check.

$$\nabla \cdot \mathbf{F} = -ik_x F_x + -ik_y F_y + -ik_z F_z \tag{6.78}$$

$$= -i\mathbf{k} \cdot \mathbf{F} = -i^2 \frac{\mathbf{k} \cdot \mathbf{k}}{|\mathbf{k}|^2} e^{-i\mathbf{k}\cdot\mathbf{r}} = e^{-i\mathbf{k}\cdot\mathbf{r}} \tag{6.79}$$

Now we may apply the divergence theorem where $\Sigma = \partial\Omega$ is the boundary of Ω.

$$S(\mathbf{k}) = \int_\Omega \nabla \cdot \mathbf{F}\, dV = \int_\Sigma \mathbf{F} \cdot d\mathbf{A} \tag{6.80}$$

$$= \sum_{n=0}^{N-1} \int_{\Sigma_i} \mathbf{F} \cdot \hat{\mathbf{n}}_\mathbf{i}\, dS_i \tag{6.81}$$

$$= \frac{i}{|\mathbf{k}|^2} \sum_{i=0}^{N-1} \mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i} \int_{\Sigma_i} e^{-i\mathbf{k}\cdot\mathbf{r}} dS_i \tag{6.82}$$

where we have decomposed the surface of the polyhedron into its facet surfaces and for which we will carry out integrals in local coordinate systems. Aligning the $\hat{\mathbf{z}}$ axis with the face normal, it is instructive to decompose $\mathbf{r}$ into vectors aligned with the local axes and distribute.

$$\mathbf{k} \cdot \mathbf{r} = \mathbf{k} \cdot \mathbf{r}_x + \mathbf{k} \cdot \mathbf{r}_y + \mathbf{k} \cdot \mathbf{r}_z \tag{6.83}$$

Note that $\mathbf{r}_z = (\mathbf{r} \cdot \hat{\mathbf{n}})\hat{\mathbf{n}} = \hat{\mathbf{z}}d$ is the distance d of the face from the origin, so

$$\mathbf{k} \cdot \mathbf{r}_z = (\mathbf{k} \cdot \hat{\mathbf{n}})(\mathbf{r} \cdot \hat{\mathbf{n}}) = (\mathbf{k} \cdot \hat{\mathbf{n}})d. \tag{6.84}$$

$$\mathbf{k} \cdot \mathbf{r}_x + \mathbf{k} \cdot \mathbf{r}_y = \mathbf{k} \cdot (\mathbf{r}_x + \mathbf{r}_y) = \mathbf{k}_{xy} \cdot (\mathbf{r}_x + \mathbf{r}_y) \tag{6.85}$$

where $\mathbf{k}_{xy} = \mathbf{k} - (\mathbf{k} \cdot \hat{\mathbf{n}})\hat{\mathbf{n}}$ is the projection of $\mathbf{k}$ into the plane of the surface. Letting $\mathbf{s} = \mathbf{r}_x + \mathbf{r}_y$ be the location on the surface,

$$\mathbf{k} \cdot \mathbf{r} = \mathbf{k}_{xy} \cdot \mathbf{s} + (\mathbf{k} \cdot \hat{\mathbf{n}})d. \tag{6.86}$$

$$S(\mathbf{k}) = \frac{i}{|\mathbf{k}|^2} \sum_{i=0}^{N-1} (\mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i}) \int_{\Sigma_i} e^{-i\mathbf{k}\cdot\mathbf{r}} dS_i \tag{6.87}$$

$$= \frac{i}{|\mathbf{k}|^2} \sum_{i=0}^{N-1} (\mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i}) e^{-i(\mathbf{k}\cdot\hat{\mathbf{n}}_\mathbf{i})d_i} \int_{\Sigma_i} e^{-i\mathbf{k}_{xy}\cdot\mathbf{s}} dS_i \tag{6.88}$$

$$= \frac{i}{|\mathbf{k}|^2} \sum_{i=0}^{N-1} (\mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i}) e^{-i(\mathbf{k}\cdot\hat{\mathbf{n}}_\mathbf{i})d_i} \int_{\Sigma_i} e^{-i(\mathbf{k}-(\mathbf{k}\cdot\hat{\mathbf{n}}_\mathbf{i})\hat{\mathbf{n}}_\mathbf{i})\cdot\mathbf{s}} dS_i \tag{6.89}$$

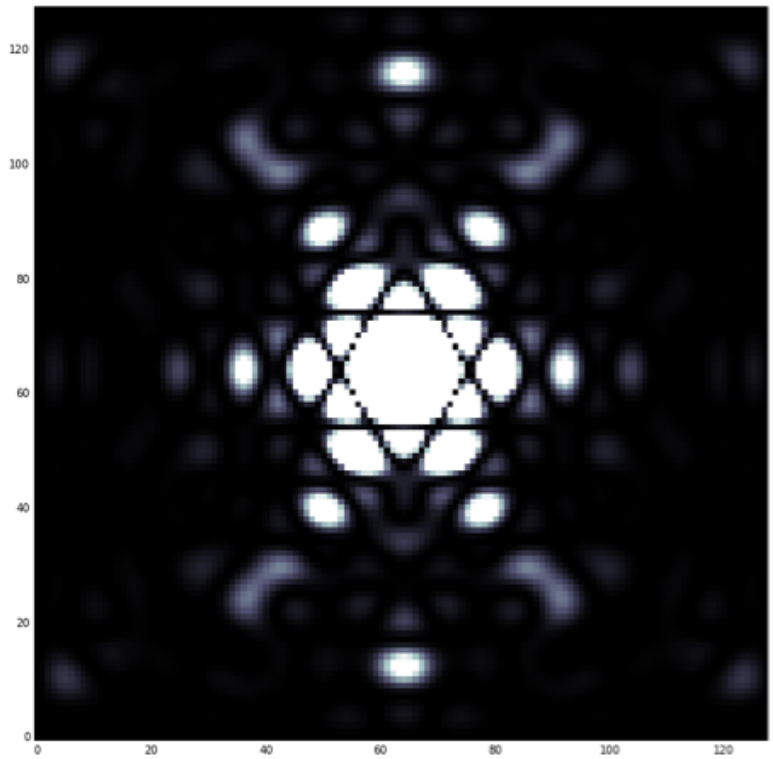


Figure 6.3: Fast Fourier Transform (discrete) of a cubic region of uniform density oriented to be viewed along the body diagonal.

The remaining integral is the Fourier transform of a polygon (6.72). Reintroducing the scattering density ρ_0, we can write the form factor of a polyhedron.

$$P(\mathbf{k}) = \frac{i\rho_0}{|\mathbf{k}|^2} \sum_{i=0}^{N-1} (\mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i}) e^{-i(\mathbf{k}\cdot\hat{\mathbf{n}}_\mathbf{i})d_i} P_i(\mathbf{k} - (\mathbf{k} \cdot \hat{\mathbf{n}}_\mathbf{i})\hat{\mathbf{n}}_\mathbf{i}) \tag{6.90}$$

P_i is the Fourier transform of the i^{th} polyhedron as given by (6.72). As before, for $|\mathbf{k}|^2$ we observe that the Fourier transform becomes trivial and $\frac{S(0)}{\rho_0} = V$, where V is the volume of the polyhedron.

Equation (6.90) is structured such that it is straight-forward to directly compute the form factor of a polyhedron with a sufficiently rich data structure that contains information on the vertices contained in the facets and the hyperplane equations of the facets.

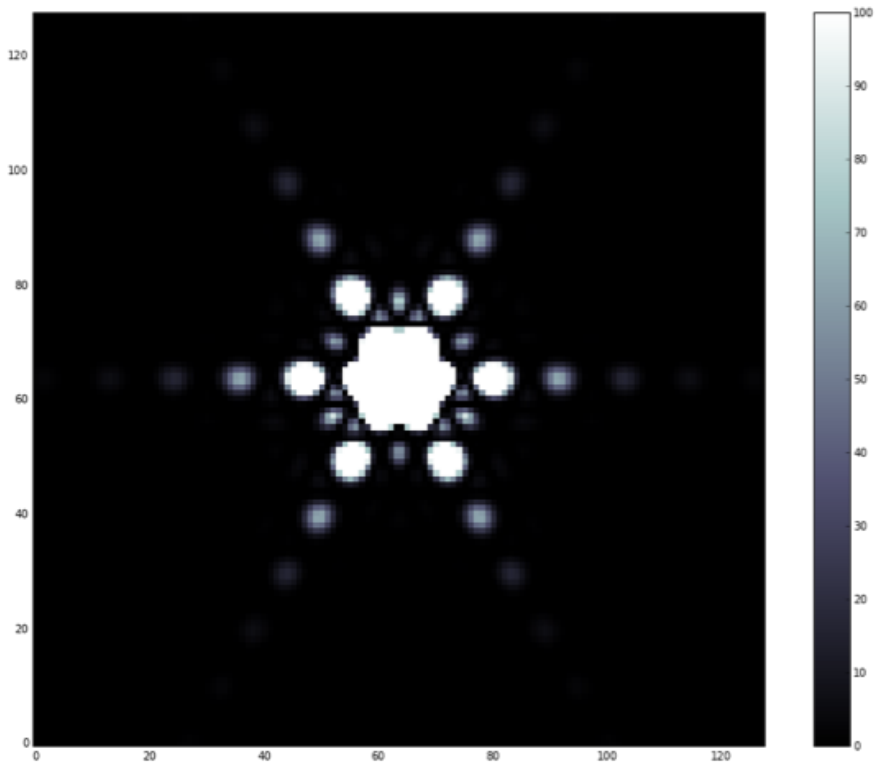


Figure 6.4: Analytic solution to cube Fourier transform shows that the discrete FT (Figure 6.3) includes many artifacts that could confuse simulated diffraction analysis.

6.4 Orientation

For anisotropic structures or particles, we would prefer to rotate our $\mathbf{k}_0$ or $\mathbf{K}$ vectors instead of the particles or structure to save on computation. Note that we may conveniently write the Fourier transform of a rotated system in terms of the Fourier transform of the non-rotated system.

6.4.1 Fourier transform under rotation

For a rotation operator $\mathbf{R}$ (an orthogonal matrix with determinant 1) let $F(\mathbf{k})$ be the Fourier transform of $f(\mathbf{x})$ and $F'(\mathbf{k})$ be the Fourier transform of $f'(\mathbf{x}) = f(\mathbf{Rx})$.

$$F(\mathbf{k}) = \int f(\mathbf{x}) \exp(-i\mathbf{k} \cdot \mathbf{x}) d\mathbf{x} \tag{6.91}$$

$$= \int f(\mathbf{x}) \exp(-i\mathbf{k}^T\mathbf{x}) d\mathbf{x} \tag{6.92}$$

$$F'(\mathbf{k}) = \int f'(\mathbf{x}) \exp(-i\mathbf{k}^T\mathbf{x}) d\mathbf{x} \tag{6.93}$$

$$= \int f(\mathbf{Rx}) \exp(-i\mathbf{k}^T\mathbf{x}) d\mathbf{x} \tag{6.94}$$

$$= \int f(\mathbf{y}) \exp(-i\mathbf{k}^T(\mathbf{R}^{-1}\mathbf{y})) d\mathbf{y}, \text{ with } y = \mathbf{Rx} \tag{6.95}$$

Note that $\mathbf{k}^T(\mathbf{R}^{-1}\mathbf{y}) = (\mathbf{k}^T\mathbf{R}^{-1})\mathbf{y} = ((\mathbf{R}^{-1})^T\mathbf{k})^T\mathbf{y} = (\mathbf{Rk})^T y$. Then

$$F'(\mathbf{k}) = \int f(\mathbf{y}) \exp(-i(\mathbf{Rk})^T y) d\mathbf{y} \tag{6.96}$$

$$F'(\mathbf{k}) = F(\mathbf{Rk}) \tag{6.97}$$

6.4.2 Handling orientation through wave vector rotation

In practice, we might intuitively think of an orientation as a rotation operation that rotates a shape from a local coordinate system to a global coordinate system. If $\rho(\mathbf{r})$ is a shape function or distribution and $F(\mathbf{K})$ its Fourier transform, let $F'(\mathbf{K})$ be the Fourier transform of the rotated shape $\rho'(\mathbf{r})$. Let $\mathbf{R}$ be the rotation operator that maps points in f to f'.

$$\rho'(\mathbf{r}) = \rho(\mathbf{R}^{-1}\mathbf{x}) \tag{6.98}$$

$$F'(\mathbf{K}) = F(\mathbf{R}^{-1}\mathbf{K}) \tag{6.99}$$

In the same way, we can rotate $\mathbf{K}$ vectors relative to the particles in the case that the system is being probed from a different angle. However, in the case of examining

a single unit cell at a pre-calculated number of k-space points, it probably makes sense to rotate the particle positions and orientations instead.

6.5 Other considerations

6.5.1 Multiple scattering

One of the assumptions in the above discussion is that scattered rays all have the same k vector before interacting with a scatterer. The relative amount of incident energy that is scattered is generally small and the chances of a probe (electron, photon, neutron, etc.) interacting twice with the sample is small if the sample is not very thick. However, in the case of the samples discussed in chapter VII, the samples are thicker than what is typical for TEM diffraction measurements and multiple scattering is almost assured. We do not present here any attempt to simulate or account for multiple scattering.

6.5.2 Complex scattering density

This author does not profess to have any particular knowledge of whatever scattering physics may be peculiar to mesoscale particles. Elastic scattering does not preclude a phase shift in scattered waves. For a single species of particle, we could neglect any such phase shift in our calculations, but for multiple particle types, differences in the phase shift from different particles would effect the scattering pattern. Brief consultation with the microscopists Xingchen Ye and Mike Katz did not yield any insight into the matter, so I chose not to assume real-valued scattering densities in a software implementation. The formulae in this chapter are similarly agnostic.

6.6 Application

I developed proofs of concept in Python and later accelerated with C++ for the computationally intensive Fourier transforms. A graphical interface, using QT libraries, allows the user to interactively adjust particle sizes, form factor, and scattering density. Both the magnitude and phase of the (complex-valued) scattering density are adjustable. The user may select from a menu any particle form factor for which Fourier transforms have been implemented. Initial versions of the tool allow delta functions, spheres of uniform density, and polyhedra (if defined in the input file). Particle positions and orientations are loaded from an input file along with a box matrix that is assumed to be periodic. Real space and reciprocal space lattice vectors are displayed as working parameters, along with the number and maximum value of the k-space coordinates that must be processed for the chosen display size and resolution.

For the session shown in Figure 6.5, peak brightness for spherical scatters was shown to be modulated radially according to the diameter and magnitude of particle scattering densities. Relative peak brightness is profoundly affected by scattering densities of different relative phases.

6.7 Outlook

This project has been on hiatus since it was deemed to be unnecessary for the completion of the project described in chapter VII, but could ultimately be of broad utility. Code now exists for performing explicit analytic Fourier transforms of particles with form factors of delta functions, uniform spheres, or polyhedra at arbitrary orientations. To be of general use, this code should be provided with effective test-

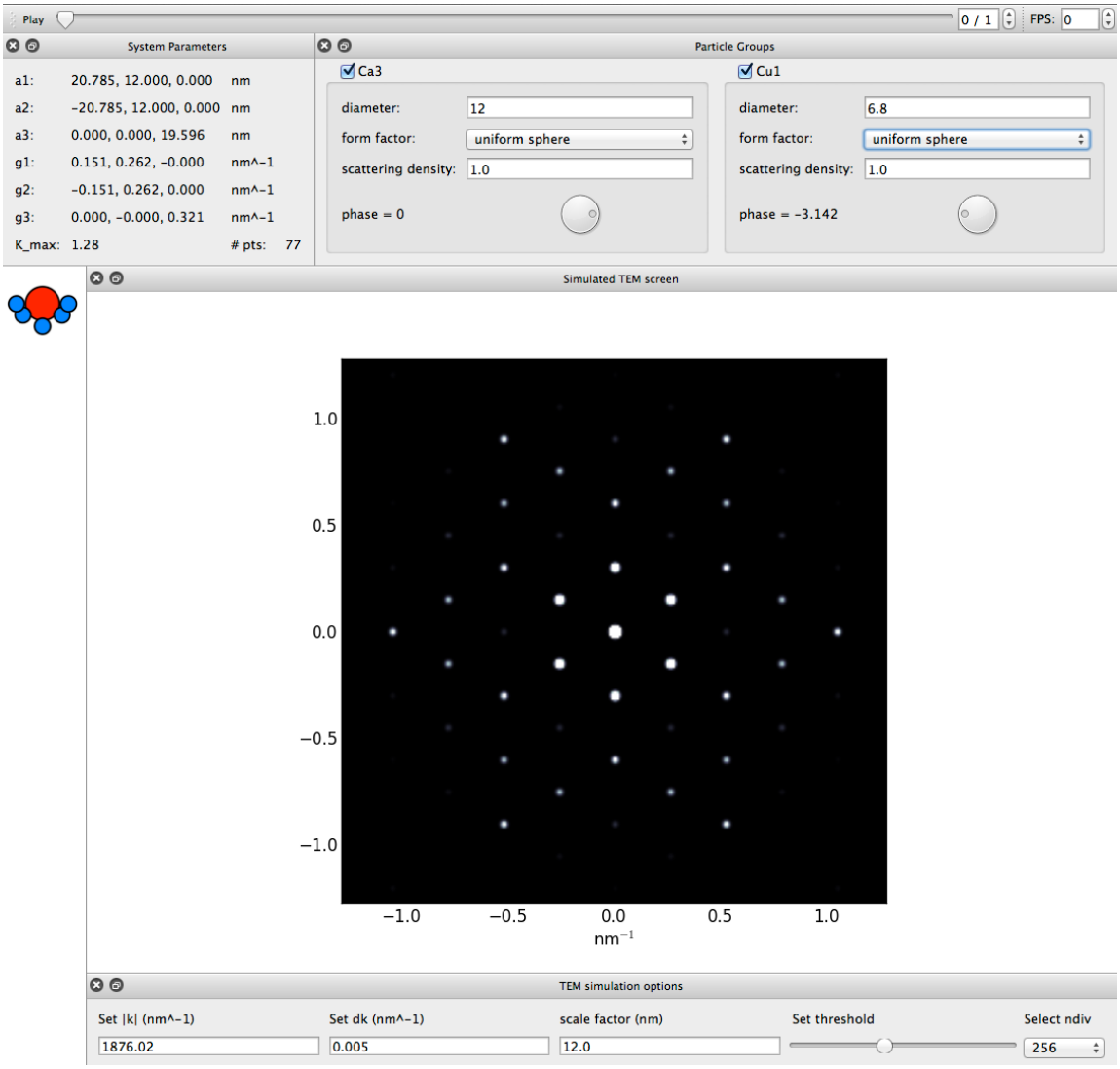


Figure 6.5: Sample interface for diffraction simulation. A single cell of a $CaCu_5$ structure (shown red and blue, mid-left) is modeled with various particle sizes, shapes, and scattering densities for comparison to experimental diffraction patterns. A particular instrument is modeled by setting the angular wave number of the "probe beam," $|k|$. For the interface shown, peak intensities are calculated for the relevant k-space points and interpolated onto a grid with a Gaussian. A faster, cleaner display consistent with graphics in literature is obtained by simply rendering circles, centered at each diffraction peak, with a diameter representative of the peak intensity.

ing and better integrated into an analysis workflow. Full integration into the Glotzer group's in-house analysis software suite (`freud`) may also require debugging of some shared data structures. Moreover, the Fourier transform code would benefit greatly from a parallel computing framework.

There are also many cases in which users may prefer not to evaluate Fourier transforms of hard particles at precise positions and orientations. If certain correlations between particle positions or orientations are known various averaging techniques may be applied to form factors without considering individual particles [92]. In general, softer form factors (i.e. non-uniform scattering densities) may be necessary [93]. Other experimental effects on diffraction peak can be modeled [92, 94], such as the Debye-Waller factor for brightness due to temperature, and the Lorentz factor for peak shape. Multiple scattering is not an insurmountable challenge [95] and could be addressed in future revisions.

Different types of output may warrant different optimizations. For instance, many x-ray diffraction measurements produce data for one-dimensional scattering angle or wave-number. Reduced k-space dimensionality reduces the computational cost of modeling the experimentally accessible data [88].

Finally, this chapter has not discussed past work in the Glotzer group to calculate structure factors through discrete Fourier transforms. Due to the efficiency of the numeric Fast Fourier Transform (FFT) method, approximate structure factors for many simulation systems can be calculated in near real time. Data transformations associated with non-cubic boxes and arbitrary viewing angles have not yet been ported from the group's `injavis` visualization software to the actively developed `freud` framework.

CHAPTER VII

Discovery, analysis, and modeling of a dodecagonal quasicrystalline nanometallic binary super-lattice

This chapter includes material being prepared for submission for publication with coauthors Xingchen Ye, Jun Chen, Michael Engel, Sharon C. Glotzer, and Chris Murray.

7.1 Introduction

Quasicrystals are aperiodic ordered solids that exhibit rotational symmetries incompatible with periodic lattices. Since the initial discovery in metallic alloys [96–98], quasicrystals have been found in a broad range of soft matter and nanoscale systems. We now know that dendritic macromolecules [99] (12 fold symmetry, tile edge length 8.2 nm), ABC star polymers [100] (12 fold, 80 nm), binary nanoparticle mixtures [101] (12 fold, 16 nm), block co-polymer micelles [102] (12 fold/18 fold, 42 nm/65 nm), tetrablock terpolymers [103] (12-fold, 40 nm), mesoporous silica [104] (12 fold, 8.5 nm), perovskite thin films [105] (12-fold, 0.69 nm), and pentameric molecules [106] (10 fold, 0.82 nm) are all capable of forming quasicrystals in non-metallic systems on a range of length scales. So far, all known non-metallic quasicrystals have n-fold axial symmetry, which means they are quasiperiodic in two dimensions and periodic in the

third. When projected along the periodic direction, axial quasicrystals correspond to two-dimensional tilings.

Understanding the stability of quasicrystals or designing a new quasicrystal requires targeting the right local order found in individual tiles as well as an appropriate arrangement of the tiles. In soft matter or nanoscale systems, where quantum mechanical effects play a minor role in the ordering of the components, local order can be controlled systematically by the presence of two competing length scales [107–109], packing constraints [62], or appropriate bond angle [110, 111]. Two extreme scenarios are distinguished: matching rules and random tilings. Matching rules enforce quasiperiodicity by energetically penalizing any incorrect tile attachment [112]. In contrast, random tilings embrace tile reshuffling and predict the emergence of a quasicrystal due to entropy maximization [113]. In practice, the growth of a quasicrystal can proceed by a rapid stochastic process followed by a slower error-and-repair process minimizing energy [114]. The structural quality will then depend on the strength of tile interactions and the available time for the repair process during growth or annealing. Matching rules are particularly important for dodecagonal (12-fold) square-triangle quasicrystals [115]. Squares and triangles easily tile the plane periodically and can form ordered or disordered tilings depending on matching rule strength [116, 117]. To achieve a well-ordered square-triangle quasicrystal, certain tile contacts or vertex configurations should be biased [118, 119]. We call this process of biasing 'partial matching rules' if the bias is incomplete or too weak to enforce quasiperiodicity without disorder. Quasicrystals with partial matching rules are expected to be better ordered than maximally random tilings.

In this work, we present a square-triangle dodecagonal quasicrystal (DQC) in

a binary mixture of nanocrystals (NCs). NCs are now routinely synthesized with sufficient monodispersity to be used as 'superatoms' for self-assembling simple or exotic superlattices [120] including quasicrystals [101,121–123]. NCs are large enough to be easily resolved via electron microscopy and sufficiently small to diffuse fast enough to achieve equilibration. Although the structure of superlattices can often be determined directly from projections using electron microscopy images, this is not always possible if the superlattice is too complex. In this case electron tomography can assist [124]. It is then possible not only to extract the unit cells of a periodic superlattice, but also to study the appearance of dislocations [125] or other defects [122] and structural variations of quasicrystals. The quasicrystal we report is of hitherto unknown structure type. We demonstrate how the decoration of the square and triangle tiles naturally gives rise to partial matching rules via symmetry breaking in layers perpendicular to the dodecagonal axis. We analyze the geometry of the experimental tiling and reproduce the growth of the binary quasicrystal via molecular dynamics simulation of a simplified model system.

7.2 Observation of quasicrystalline binary NC superlattice (BNSL)

The DQCs were formed by slow drying of a binary NC solution on top of an immiscible liquid subphase. Figure 7.1a shows a large-area TEM image of binary nanoparticle super lattices (BNSLs) self-assembled from 6.8 nm $CoFe_2O_4$ and 12.0 nm Fe_3O_4 NCs. The BNSLs appear to possess long-range order, yet no translational periodicity can be found. Small-angle electron diffraction (SAED) pattern is characterized by a set of sharp diffraction spots with a clear 12-fold rotational symmetry (Fig. 7.1b), which is the characteristic signature of a DQC and indicates that the structural perfection persists over length scales greatly exceeding prior observations

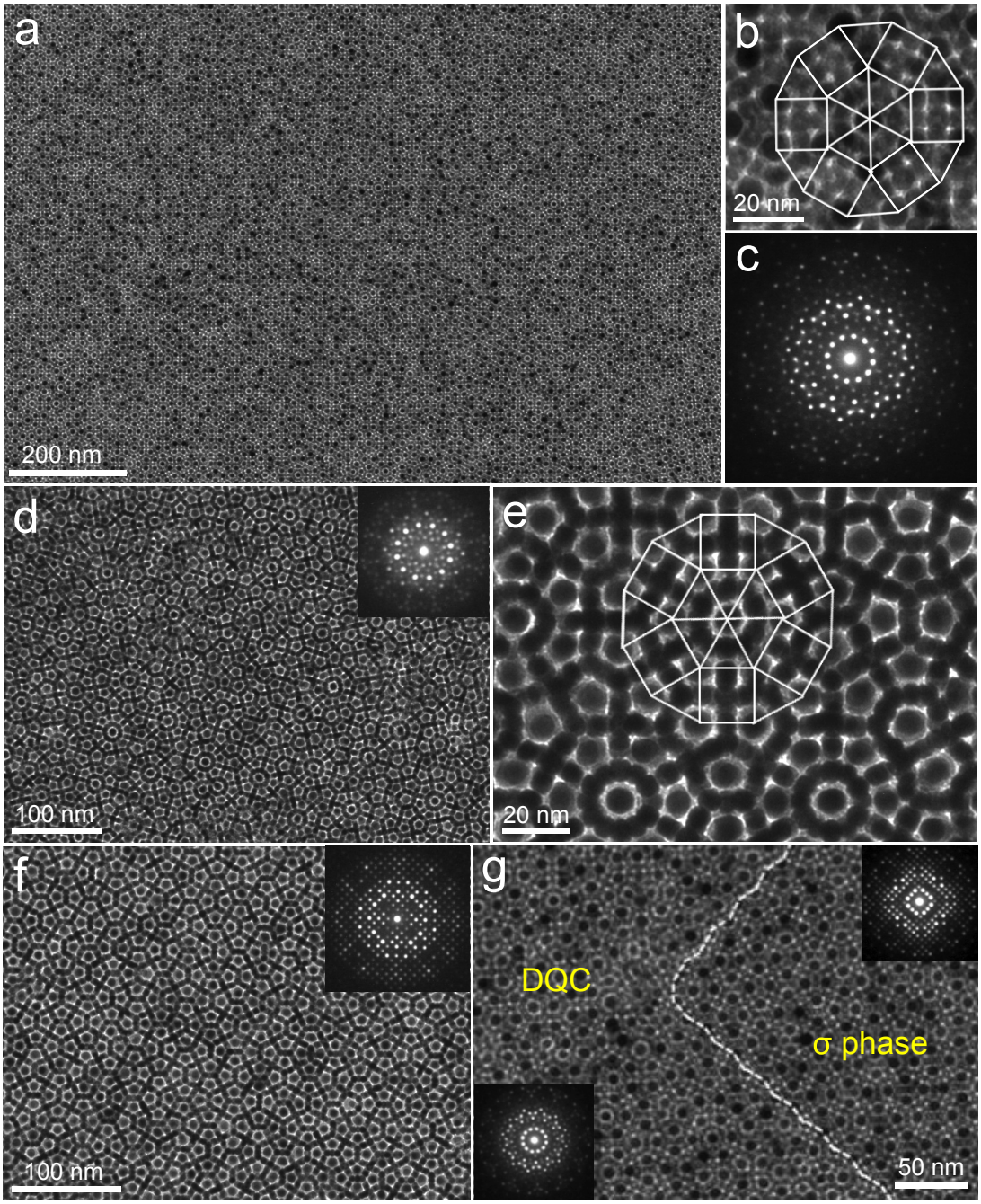


Figure 7.1: Self-assembled BNSLs with quasiperiodic and periodic order. (a-c) Low-magnification TEM image (a), SAED pattern (b) and high-magnification TEM image of dodecagonal quasicrystalline BNSLs self-assembled from 6.8 nm $CoFe_2O_4$ and 12.0 nm Fe_3O_4 NCs. (d) Low-magnification TEM image and SAED pattern (inset) and (e) high-magnification TEM image of dodecagonal quasicrystalline BNSLs self-assembled from 6.2 nm FePt and 11.5 nm Fe_3O_4 NCs. (f) TEM image and SAED pattern (inset) of competing periodic Frank-Kasper σ-phase in FePt–$CoFe_2O_4$ BNSLs. (g) TEM image and SAED patterns (insets) showing the coexistence of quasicrystalline and the Frank-Kasper σ-phase in $CoFe_2O_4$–Fe_3O_4 BNSLs.

of DQCs in BNSLs. High-magnification TEM imaging shows that the DQCs can be described by a self-similar aperiodic arrangement of square and triangle tiles (Fig. 7.1b). In a common motif, the squares and triangles form dodecagons by having six triangles in the center, surrounded by an alternating arrangement of six squares and triangles, giving rise to a 12-fold rotational symmetry. The same DQC phase was also realized in BNSLs composed of FePt and Fe_3O_4 NCs (Figs.7.1d, e), Fe_3O_4 and Fe_3O_4 NCs and 5.8 nm Au and 9.7 nm Fe_3O_4 NCs, provided that the NC size ratio lies in the range of 1.51–1.67. The NC size is computed as the sum of the inorganic core diameter plus the ligand shell thickness. These results provide compelling evidence that the formation of DQC is robust and does not depend significantly on details of the constituent NCs such as their chemical composition. In line with previous studies [101, 121–123], the present DQC can coexist with its approximant, the so-called Frank-Kasper σ-phase (Fig. 7.1f), which is observed to grow alongside DQCs in the FePt-Fe_3O_4 and the $CoFe_2O_4$-Fe_3O_4 systems (Fig. 7.1g). The σ-phase is a periodic (3^2.4.3.4) Archimedean square-triangle tiling, and is indistinguishable from DQC locally, but differs in global ordering. Other periodic BNSLs of the $NaZn_{13}$-type and the $CaCu_5$-type were also found to coexist with DQCs and the σ-phase, though these phases occurred less frequently.

7.3 Electron microscopy and tomographic reconstruction

Although the tiling we observe is the common square-triangle tiling, the decoration of the tiles is distinct from previous reports of quasicrystals in binary NC BNSLs. Its structure is more complex and cannot easily be solved by direct visual inspection of TEM images. We therefore use scanning electron microscopy (SEM), which allows us to image the surface termination, as well as tomographic reconstruction from TEM

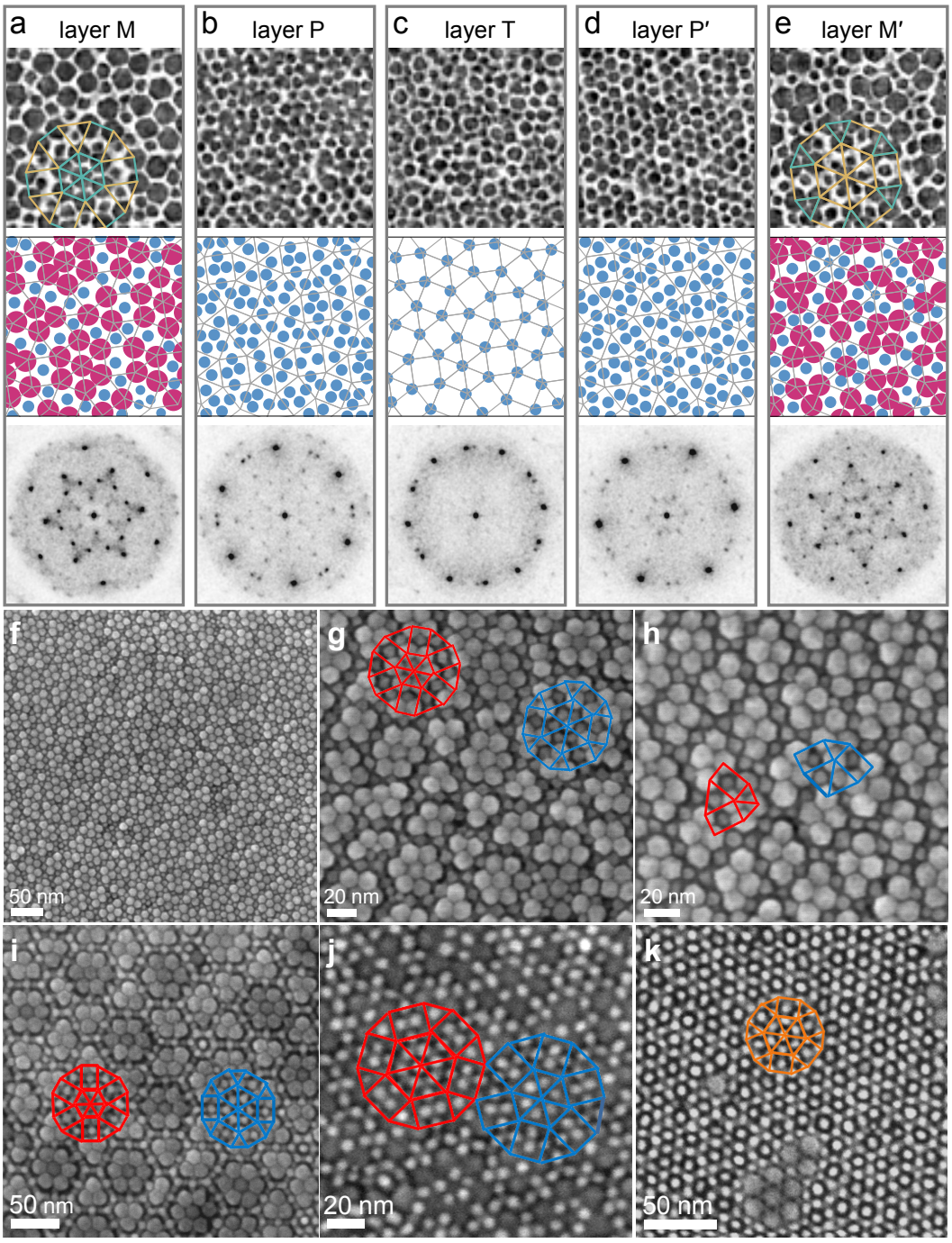


Figure 7.2: Tomographic reconstruction and SEM imaging reveals the 3D structure of quasicrystalline BNSLs. (a-e) Horizontal reconstruction slices showing the NC arrangement within layers M, P, T, P′, M′ at different heights. (a-e)Tomographic slices (top row), corresponding decorated square-triangle tiling (middle), and the Fourier transform pattern of each tomographic slice(bottom) are shown. Except for layer T, all other layers exhibit six-fold rotational symmetry. (f-h) SEM image of quasicrystalline (f,g) and σ-phase (h) $CoFe_2O_4$–Fe_3O_4 BNSLs. (i-k) SEM image of quasicrystalline FePt–Fe_3O_4 BNSLs. Three distinct surface terminations of the quasicrystalline phase were observed from different BNSL domains.

for a full three-dimensional structure analysis. SEM of $CoFe_2O_4$–Fe_3O_4 samples reveals that the surface termination in the quasicrystal (Fig. 7.2f,g) and the σ-phase (Fig. 7.2h) is dominated by large NCs packed together closely in triangles and separated by small NCs sitting slightly lower. The large NCs do not sit exactly at tile vertices, as is visible by connecting their positions with lines. The resulting tiling consists of small triangles, large triangles, and rectangles, and is distinct from the square-triangle tiling that emerges in projection in the TEM images. Red and blue overlays in the figure show how each motif in each system can be constructed two ways by swapping large and small triangles while rotating rectangles by 90 degrees.

The complete crystallographic structure of the BNSLs can be obtained using electron tomography. In this technique TEM projections are recorded from various directions and, with the help of tracer NCs, three-dimensional structure information reconstructed. The resulting tomographic data is of high enough resolution to resolve the positions and type (large or small) of most individual NCs. Though NCs are not well discriminated in the perpendicular axis in the tomographic data, we can clearly identify layers in which the centers of NCs are approximately located. Slices through the middles of the layers of a quasicrystal BNSL sample are depicted in Fig. 7.2a–e (top). Note that interpenetration of layers and z-axis blurring cause NCs to also appear in neighboring slices.

Next, by visual inspection, we show only NCs whose centers lie in roughly the same plane (Fig. 7.2a–e (middle)). Depending on the tile decoration in the layers, and for reasons discussed below, we distinguish mirror (M, M′), puckered (P, P′), and tile (T) layers. The sequence of panels illustrates that the spots of low transmittance in the TEM images defining the square-triangle tiling vertices correspond to stacks of

large NCs. We also see that the different surface terminations imaged in Fig. 7.2i–k correspond to the M–P–T sequence.

The symmetries of the layers are easily seen in Fourier transforms of the full tomography slices (Fig. 7.2a–e (bottom)). Features of interest are enhanced with smoothing and contrast by applying a short-range Gaussian blur and extreme value clipping. Interestingly, the 12-fold symmetry of the overall structure is broken into 6-fold symmetry in all layers except the T layer. The orientation of the 6-fold symmetric layers rotates by 30 (or 90) degrees between the non-primed (P, M) and primed (P′, M′) layers. Such a symmetry breaking has recently also been observed in the hard-core/square-shoulder model [126].

We should reiterate that the Fourier transforms shown in Fig. 7.2a–e (bottom) are from a much larger sample area than that shown in the real space images. The full sample contains small areas of other structures that contribute a six-fold pattern, seemingly breaking the twelve-fold symmetry of the T layer. We note that the particle correlations at the orientation and length scale indicated by the dimmer six-fold peaks are incommensurate with the square-triangle tiling contributing the strong peaks and clearly can not be attributed to the same structure.

7.4 Structure solution

We solve the geometric decoration of the square and triangle tiles with large and small NCs. As is clear from this structure model shown in Fig. 7.3a, mirror layers (M, M′) are in fact mirror planes, puckered layers (P, P′) are not perfectly planar but slightly 'puckered', and the tile layer (T) contains small particles at the tile vertices. For square tiles, the tiling layer contains a local roto-inversion symmetry in the tile center mapping M on M′ and P on P′. Square tiles contain 2 large and 16 small NCs

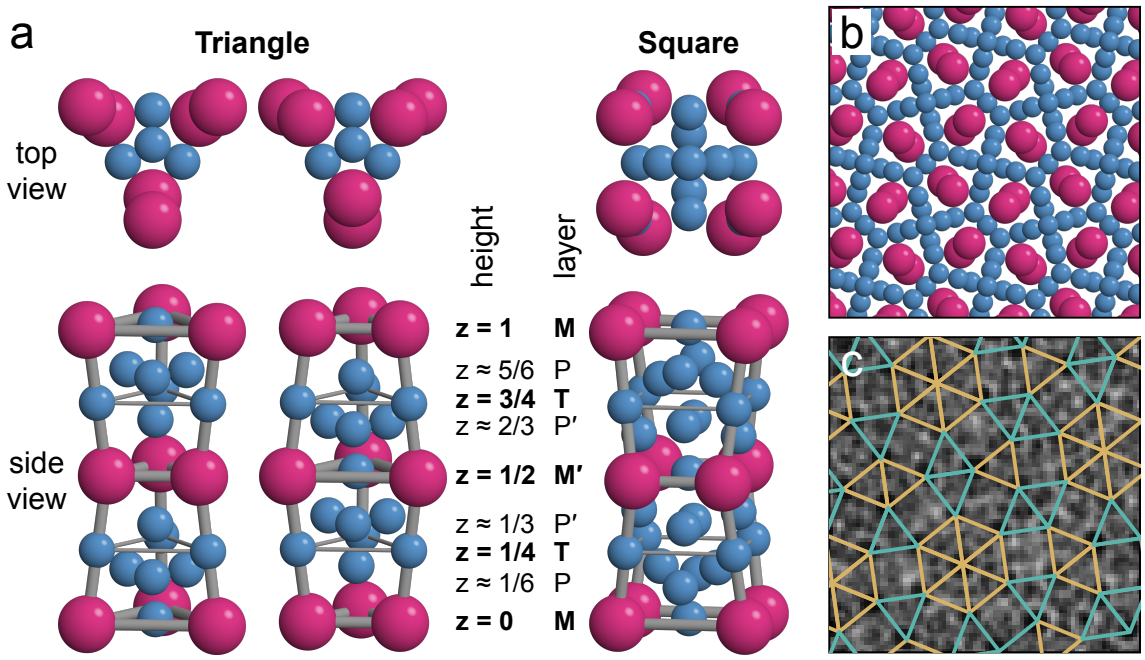


Figure 7.3: Binary nanoparticle super-lattice structure model. (a) The square-triangle tiling consists of two symmetry-equivalent triangle tiles and a square tile. NCs align on flat mirror layers (M, M′), flat layers that contain small particles at tile vertices (T), and puckered layers (P, P′). (b) In the σ-phase the tile decoration slightly adjusts to the local symmetry. (c) In the M and P layers the twelve-fold symmetry of the quasicrystal is broken, which is possible because the edges of square-triangle tilings are bipartite, as highlighted here in the tiling with two colors. For (c) we averaged the tomographic slices of Fig. 7.2g–k to demonstrate that twelve-fold symmetry is restored in projection.

(AB_8). Triangle tiles contain 1 large and 7 small NCs (AB_7). The observed σ-phase unit cell then contains 8 large and 60 small NCs ($AB_{7.5}$). For a full sequence M, P, T, P′, M′, P′, T, P, M of height z = 1 and M′ at z = 1/2, the puckered and tile layers are not evenly spaced vertically in the cell but lie at approximately z = 2/12, 3/12, 4/12 and z = 8/12, 9/12, 10/12. This is evident both from packing considerations and by examination of the tomographic data. The large NCs occurring in M and M′ layers form staggered columns. Lines connecting large NCs to neighbors in the same layer provide tile edges of two lengths. Long and short edges alternate from M to M′ layers so that a square tile consists of layers of rectangles with alternating orientations and triangular tiles contain layers of alternating large and small triangles. Based on the structure model we postulate a point group 12/mmm and five-dimensional space group P 12_6/m 2/c 2/m.

The square and triangle decorations shown in Fig. 7.3a are developed geometrically with maximal symmetry. Some symmetry is lost with the internal strain necessary for different neighbor relationships. For instance, in the σ-phase, the alternating layers of large NCs lie along the shared edge of two triangles rather than shifting radially from the tile center (Fig. 7.3b). Depending on the environment, one or more columns of large NCs may be pinned to a tile vertex (as in a hexagon of six triangle tiles) producing internal strain within the tile, where large NCs would otherwise be staggered within the column between layers. In some instances, the strain can be resolved through defects (e.g. the replacement of a large NC by a small one), which we observe frequently in the electron tomography data. The distribution of the two edge types (long and short) in the M and M′ layers is governed by two rules: (1) the edges in a triangle are of the same type; (2) the edges in a square alternate

in type. As can be shown, these rules are applicable to any square-triangle tiling. The result is a grouping of edges into short and long based on their orientation alone (Fig. 7.3c). This explains the breaking of the 12-fold symmetry to 6-fold symmetry observed in individual M and P layers. 12-fold symmetry is observed only in the T layer or when averaging all layers by projection.

The unusual decoration of the square and triangle tiles affects the tiling geometry. Joining two triangles or two squares along an edge or more than two triangles without the separation of a square around a vertex introduces internal stress, and thus an energetic penalty in the form of partial matching rules. The matching rules are not strict, because the dodecagonal quasicrystal BNSL has the ability to absorb a certain amount of that stress as is visible from the presence of various local tile neighborhoods. Still, we expect that the presence of the partial matching rules means our dodecagonal quasicrystal will be less random than without it. Such randomness can be quantified by analyzing the phason displacement field of the quasiperiodic tiling.

7.5 Image processing and tiling analysis

7.5.1 Analysis of tomography data

Using code developed for tracking ellipsoidal colloids in confocal microscopy data using watershed cuts [127], we attempted to identify particle positions and orientations in the tomography data to confirm the structural model and help refine an anisotropic particle model. Noise, particularly in the z-direction, inhibited our ability to effectively localize particles and we were not able to automatically locate particles with sufficient accuracy in three dimensions (3d). However, we were able to develop an effective workflow to identify and analyze two-dimensional (2d) features from the

TEM images.

7.5.2 Extraction of tile vertices

Dark spots of relatively uniform size and shape in the TEM images mark tile vertices. The sequence of panels in figure 7.2 illustrates that the stacks of large particles correspond to the spots of low transmittance in the TEM images as in the structural model. By convolving the image with a disk of the same size, the centers of the spots are accentuated. A threshold separates the peaks, then a watershed cut is applied to build clusters of pixels associated with each local maximum. Centers of clusters serve as candidate vertices. The 2d vertex coordinates so determined serve as the input for quantitative analysis of the tilings. The automated tile vertex detection worked well for $CoFe_2O_4$ – Fe_3O_4 systems where the vertices were notably darker in TEM. The small FePt particles in other systems have a higher electron scattering density and the resulting TEM has lower contrast between columns of particles. We attempted to work around this by identifying the ring of high electron transmission immediately around the tile vertices, but convolving an image with an annulus was not effective at extracting the inconsistent and irregular rings. Automated image processing plus some manual touch-up work has been effective at extracting as many as 15,000 vertices (from lowest magnification images) of three $CoFe_2O_4$ – Fe_3O_4 samples, referred to hereafter as samples A, B, and C.

7.5.3 Lifting of tile vertices

The quasicrystals we observe are described as tilings of squares, triangles, and thin 30 degree rhombi, which occur with low probability. Tiles have edges of uniform length a at one of twelve orientations throughout a single quasicrystalline grain.

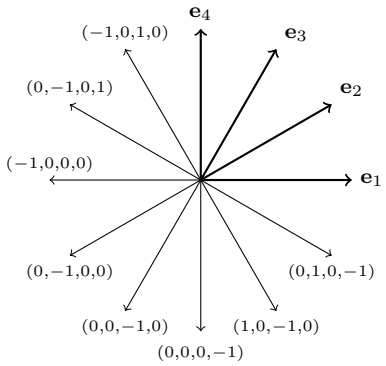


Figure 7.4: The twelve possible nearest neighbor directions can be addressed in terms of four (quasi) lattice vectors. The basis vectors referred to in this article are shown $\mathbf{e}_1$ through $\mathbf{e}_4$. Relative coordinates in terms of this basis are given for the remaining eight directions.

Only four of the twelve edge-defining vectors are linearly independent. We describe the relative coordinates of vertices along a path of edges as an integer linear combination of the four basis vectors (Fig. 7.4) The four two-dimensional vectors can be understood as projections of lattice vectors in four dimensional space onto a two-dimensional plane. The nature of the four dimensional lattice and the slope of the plane define the possible tile shapes. It has been shown that projecting the $\mathbb{D}_4$ lattice as shown below can represent all tilings of squares, triangles, and thin rhombs with quasiperiodicity when tile vertices are selected so as to minimize the distance from a plane parallel to the plane of projection.

A convenient transformation from four lattice vectors to Cartesian coordinates is given by [128]

$$\mathbf{M} = \left[\begin{array}{c} \mathbf{M}^{\parallel} \\ \hline \mathbf{M}^{\perp} \end{array}\right] = 2^{-1/2} a' \left[\begin{array}{cccc} 1 & \cos(\pi/6) & \cos(2\pi/6) & 0 \\ 0 & \sin(\pi/6) & \sin(2\pi/6) & 1 \\ \hline 1 & \cos(5\pi/6) & \cos(10\pi/6) & 0 \\ 0 & \sin(5\pi/6) & \sin(10\pi/6) & 1 \end{array}\right] \tag{7.1}$$

where a' is the higher dimensional lattice constant.

The matrix $\mathbf{M}$ transforms integer lattice coordinates in $\mathbb{D}_4$ to space in which two dimensions lie in the projection plane and two are perpendicular. The horizontal line in the matrix is a visual aid to separate the parallel space from the perpendicular space. Since this paper is primarily concerned with parallel and perpendicular space projects of $\mathbb{D}_4$, we will conveniently choose $a' = a\sqrt{2}$ with $a = 1$ the normalized tile edge length.

For a four-dimensional coordinate, then, applying the transform $\mathbf{M}$ gives a four-dimensional vector, the first coordinates of which are in parallel space and the latter two of which are in perpendicular space. E.g.

$$\{\mathbf{v}^{(2)\parallel}, \mathbf{v}^{(2)\perp}\} = \mathbf{M}\mathbf{v}^{(4)\mathbb{D}_4} \tag{7.2}$$

It is convenient to map four lattice vectors between two and four dimensions as follows.

$$(1,0) = \mathbf{e}_1^{(2)} \leftrightarrow \mathbf{M}\mathbf{e}_1^{(4)} = \mathbf{M}(1,0,0,0) \tag{7.3}$$

$$(\cos\frac{\pi}{6}, \sin\frac{\pi}{6}) = \mathbf{e}_2^{(2)} \leftrightarrow \mathbf{M}\mathbf{e}_2^{(4)} = \mathbf{M}(0,1,0,0) \tag{7.4}$$

$$(\cos\frac{2\pi}{6}, \sin\frac{2\pi}{6}) = \mathbf{e}_3^{(2)} \leftrightarrow \mathbf{M}\mathbf{e}_3^{(4)} = \mathbf{M}(0,0,1,0) \tag{7.5}$$

$$(0,1) = \mathbf{e}_4^{(2)} \leftrightarrow \mathbf{M}\mathbf{e}_4^{(4)} = \mathbf{M}(0,0,0,1) \tag{7.6}$$

7.5.4 Tile identification

The identified vertices are lifted onto the four-dimensional $\mathbb{D}_4$ lattice by assigning each vertex a four-dimensional lattice point $\mathbf{x}_j^{4\mathrm{d}}$. Assigning coordinates (0,0,0,0) to an arbitrary point in the 2d input data, a breadth first search is performed on neighboring points, assigning 4d coordinates as possible, constrained by user-definable tolerances. Each four-dimensional lattice point is then projected onto (i)

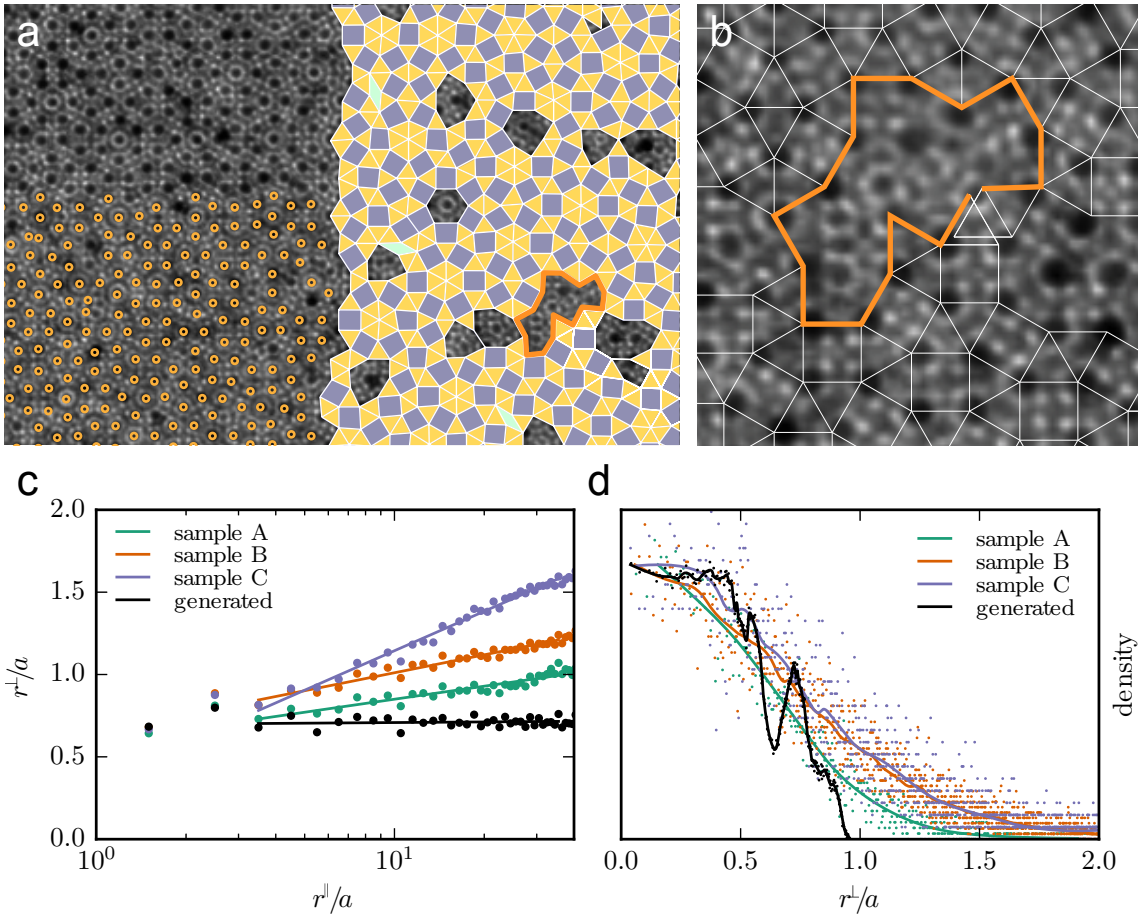


Figure 7.5: Geometric analysis of the dodecagonal quasicrystal BNSL. (a) Columns of large particles are identified with image processing (bottom left) and mapped to vertices of a square-triangle tiling (right). The mapping is hindered by the occasional presence of a dislocation (highlighted in orange). (b) The Burgers vector around the dislocation can be identified in the idealized coordinates by performing a Burgers circuit around the dislocation core. (c) We embed the tiling in four-dimensional space and analyze it for the presence of phason displacement by correlating distances in parallel space $r^\parallel$ with distances in perpendicular space $r^\perp$ in units of tile edge length a. Different samples show varying amounts of phason displacement. The data shows a linear increase of phason displacement, as expected for two-dimensional quasicrystals [129]. (d) The radial density of occupation domains in perpendicular space is smoothed out for experimental samples relative to the generated tiling.

a two-dimensional parallel space, $\mathbf{x}_j^{\parallel} = \mathrm{P}^{\parallel}\left(\mathbf{x}_j^{4\mathrm{d}}\right)$, describing the ideal, i.e. phonon strain-corrected position of the vertex, as well as (ii) onto the two-dimensional perpendicular space, $\mathbf{x}_j^{\perp} = \mathrm{P}^{\perp}\left(\mathbf{x}_j^{4\mathrm{d}}\right)$, quantifying the amount of phason displacement, i.e. the deviation from ideal quasiperiodic order.

The resulting 4d coordinates can then be projected back to 2d to compare with the input data. Nearest neighbor distances could be seen to be closer along one than another of more or less perpendicular axes, so we attempt to correct for small tilts of the imaged sample by fitting a linear transform. Optimizing input stretch, shear and rotation, through user interaction and linear regression on the transform between input and output coordinates, we are able to detect large areas of tile vertices for which the ideal 4d coordinate mapping well describes the input data.

However, actual strain in the sample is not distinguishable from apparent strain from sample alignment in our simple matrix fitting. Thus the optimal transform obtained varies depending on the order in which the network is explored and how many candidate coordinates are so obtained. In general, we find the largest detectable net with the lowest error occurs for nets initiated near the middle of the image. The lifting procedure is robust and not sensitive to point defects. Overall, it works well in our samples except for the presence of occasional dislocations (Fig. 7.5b).

For visualization purposes and for tile statistics, we identify as square, triangle, and rhomb tiles any closed loop of three or four points. Once tiles are identified and categorized, polygons can be drawn and colored using either the ideal coordinates or the original input coordinates.

7.5.5 Comparing idealized and measured coordinates

Projecting the 4d coordinates onto perpendicular space allows analyzing the quality of a tiling and distinguishes quasicrystals and approximants. Under the projection the tile vertices fall into the so-called occupation domain [130]. In agreement with dodecagonal quasicrystals found in alloys [131], typical occupation domains of our experimental samples are compact but thermally broadened (Figure 7.6). We do not observe a faceting of the occupation domain or fractal behavior, which occurs in ideal dodecagonal square-triangle tilings generated by inflation schemes as in Ref. [128] (Figure 7.6 versus Figure 7.7).

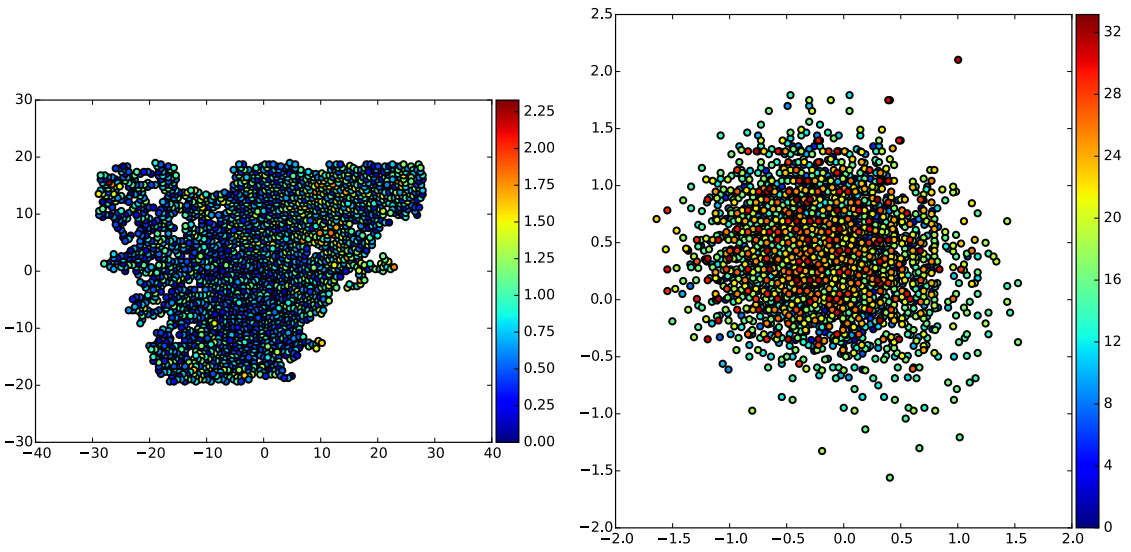


Figure 7.6: Experimental tiling in parallel (left) and perpendicular space (right), colored by distance of lattice points in the other two dimensions from a reference point.

7.5.6 Presence of dislocations

Some parameters, particularly the starting point of the network exploration, and the presence of defects have a strong effect on the lifting procedure, limiting our analysis (Figure 7.17). The most pronounced defects, predicted by quasicrystal growth

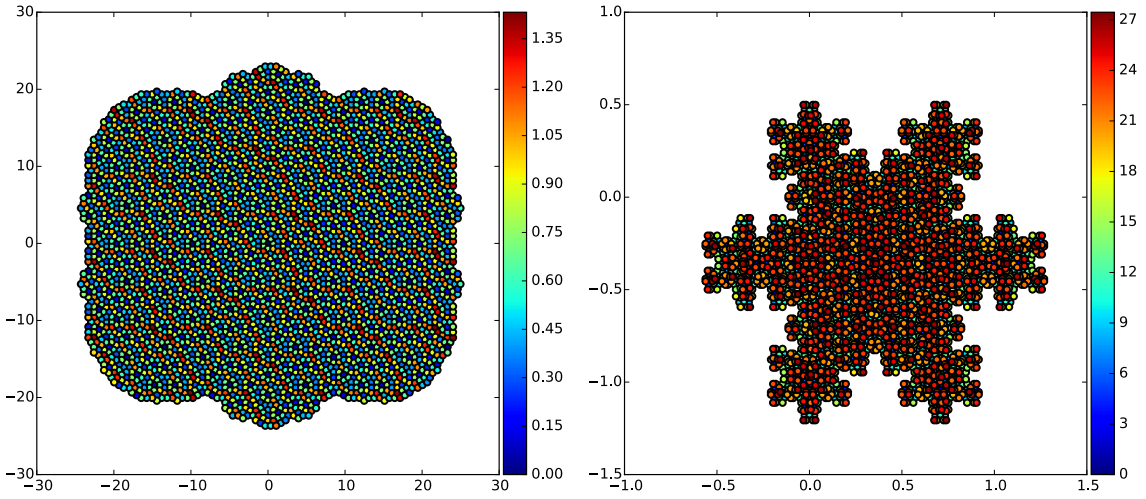


Figure 7.7: Generated tiling in parallel (left) and perpendicular space (right), colored by distance of lattice points in the other two dimensions from a reference point.

models [119], are dislocations. As the network exploration proceeds past dislocations, strain perpendicular to the direction of exploration is produced by a Burgers vector. Dislocations can then be located and characterized by examining several nets (produced from different starting points). Each dislocation causes a string of detected coordinates to disagree by the Burgers vector b (Figures 7.8, 7.14 – 7.17), which is a 4d vector that maps onto a fractional and small vector in parallel space $\mathbf{b}^{\parallel} = M^{\parallel}\mathbf{b}$ (phonon component) and a typically larger vector in perpendicular space $\mathbf{b}^{\perp} = M^{\perp}\mathbf{b}$ (phason component). We call the disagreements *lifting continuity violations* and analyze their frequency in three TEM samples. In Table 7.1 Burgers vectors are grouped by symmetry equivalence. In agreement with prior works [125] the dislocations we observe are predominantly of one type, though the Burgers vector we observe most frequently is different. The Burgers vector type (0,1,-2,1), which corresponds to the smallest phonon component in our samples, is found most frequently in our samples. Additional figures are included at the end of this chapter to show results from the network exploration of all three samples.

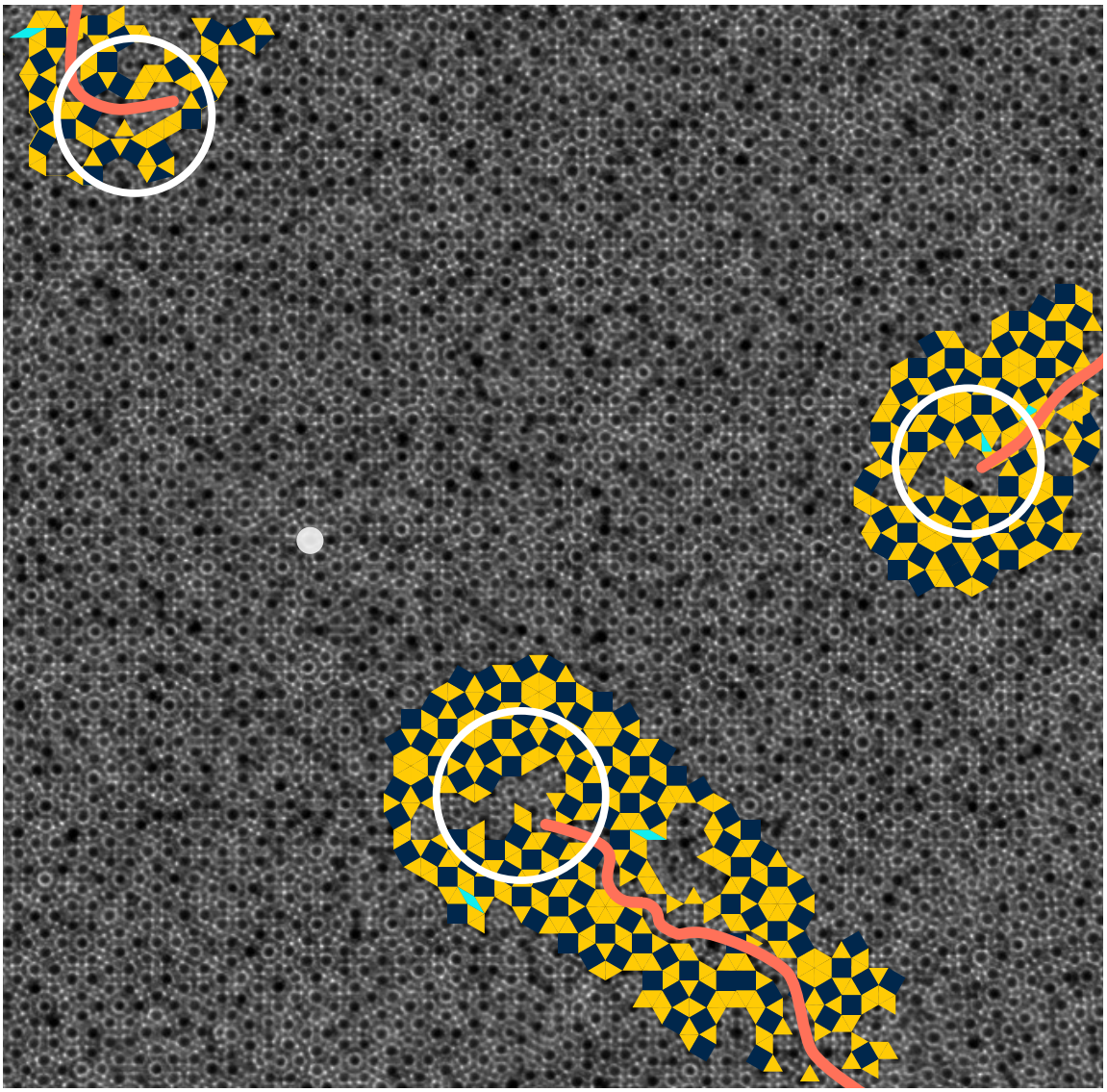


Figure 7.8: Burgers vectors are revealed during network exploration. The white dot in the center left shows the tiling vertex from which automated network exploration initiated. As exploration proceeded around the noted dislocations, the Burgers vectors produced disagreeing tile coordinates. The strain in the sample complicates coordinate detection or phason strain analysis. We merely note the path of irreconcilabity in the lattice exploration.

Table 7.1: Statistics of lifting continuity violations in three $CoFe_2O_4$–Fe_3O_4 samples. Dislocations with small phonon or small phason component are preferred (emphasis).

Burgers vector (type)	Phonon component	Phason component	Number detected
(0,1,-2,1)	*0.268*	3.73	*58*
(2,-3,1,1)	0.379	5.28	4
(0,0,-1,1)	0.518	*1.93*	*12*
(0,1,-2,1)	0.78	4.63	2

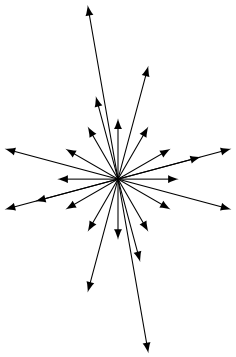


Figure 7.9: Schematic of all of the Burgers vectors observed across the three samples, normalized to the coordinate system shown in Figure 7.4

7.5.7 Phason analysis

We perform a phason displacement analysis of the TEM images. Such an analysis was previously performed for an icosahedral quasicrystal formed in a computer simulation [132] and for a dodecagonal quasicrystal found in a Mn-Cr-Ni-Si alloy [131]. We focus on $CoFe_2O_4$-Fe_3O_4 samples, in which image analysis is reasonably effective to automatically identify the tiling. Due to the strain introduced by dislocations, we identify dislocation cores and do our best to limit our phason analysis to a convex region not including any dislocation.

The phason displacement analysis correlates the distances of two vertices in parallel space, $r_{jk}^{\parallel} = \left|\mathbf{x}_j^{\parallel} - \mathbf{x}_k^{\parallel}\right|$, and perpendicular space, $r_{jk}^{\perp} = \left|\mathbf{x}_j^{\perp} - \mathbf{x}_k^{\perp}\right|$. We analyze three samples and compare them with an ideal square-triangle quasicrystal constructed via an inflation rule [128] (Fig. 7.5c). The comparison shows that our experimental samples deviate slowly from the ideal tiling and follow the behavior expected for a two-dimensional random tiling [113], $r^{\perp} = K^{-1} \ln\left(r^{\parallel}/a\right) + C$ with a phason elastic constant K, the tile edge length a, and an offset C. Apparently, phason fluctuations vary in our samples and are lowest in sample A. The comparison of the distribution of $r_{jk}^{\perp}$ with the ideal tiling (Fig. 7.5d) provides information

about the compactness of the occupation domain. As expected, phason fluctuations smoothen and slightly broaden the occupation domain boundary.

Given two 4d coordinates $\mathbf{t}_i^{(4)}$ and $\mathbf{t}_j^{(4)}$, the parallel space separation $\mathbf{r}_{ij}^{\parallel}$ and perpendicular space separation $\mathbf{r}_{ij}^{\perp}$ can be determined, where $\mathbf{r}_{ij}^{\parallel} = \mathbf{M}^{\parallel}\left(\mathbf{t}_i^{(4)} - \mathbf{t}_j^{(4)}\right)$ and $\mathbf{r}_{ij}^{\perp} = \mathbf{M}^{\perp}\left(\mathbf{t}_i^{(4)} - \mathbf{t}_j^{(4)}\right)$. The relationship $r^{\perp}\left(r^{\parallel}\right) = \left\langle r_{ij} \left| r_{ij}^{\parallel} \approx r^{\parallel} \right. \right\rangle_{ij}$, obtained by averaging over pairs, is called phason displacement. Lower values correspond to a more compact occupation domain. At longer distances, phason displacement increases and dislocations create strain. We therefore limit our analysis to connected regions of moderate uncorrected phonon strain. Such regions have at most a few thousand points. Although the growth of phason displacement with $r^{\parallel}$ depends on the starting point of the tile vertex detection, the general shape of the curve remains unchanged (Figure 7.10). We therefore choose the centermost input point as the starting vertex in Figure 7.5.

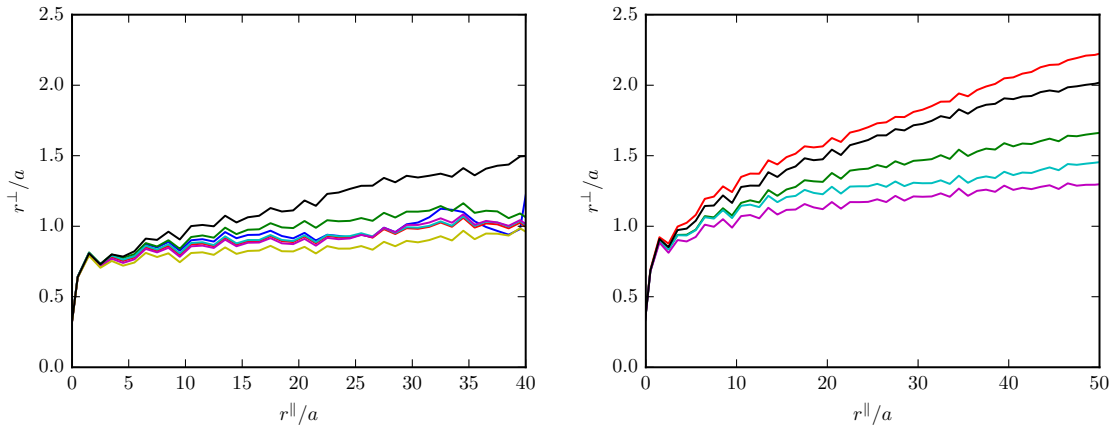


Figure 7.10: Phason displacement analysis of the (smaller) sample A (left) and the (larger) sample B (right). Each curve shows the analysis starting from a different tile vertex.

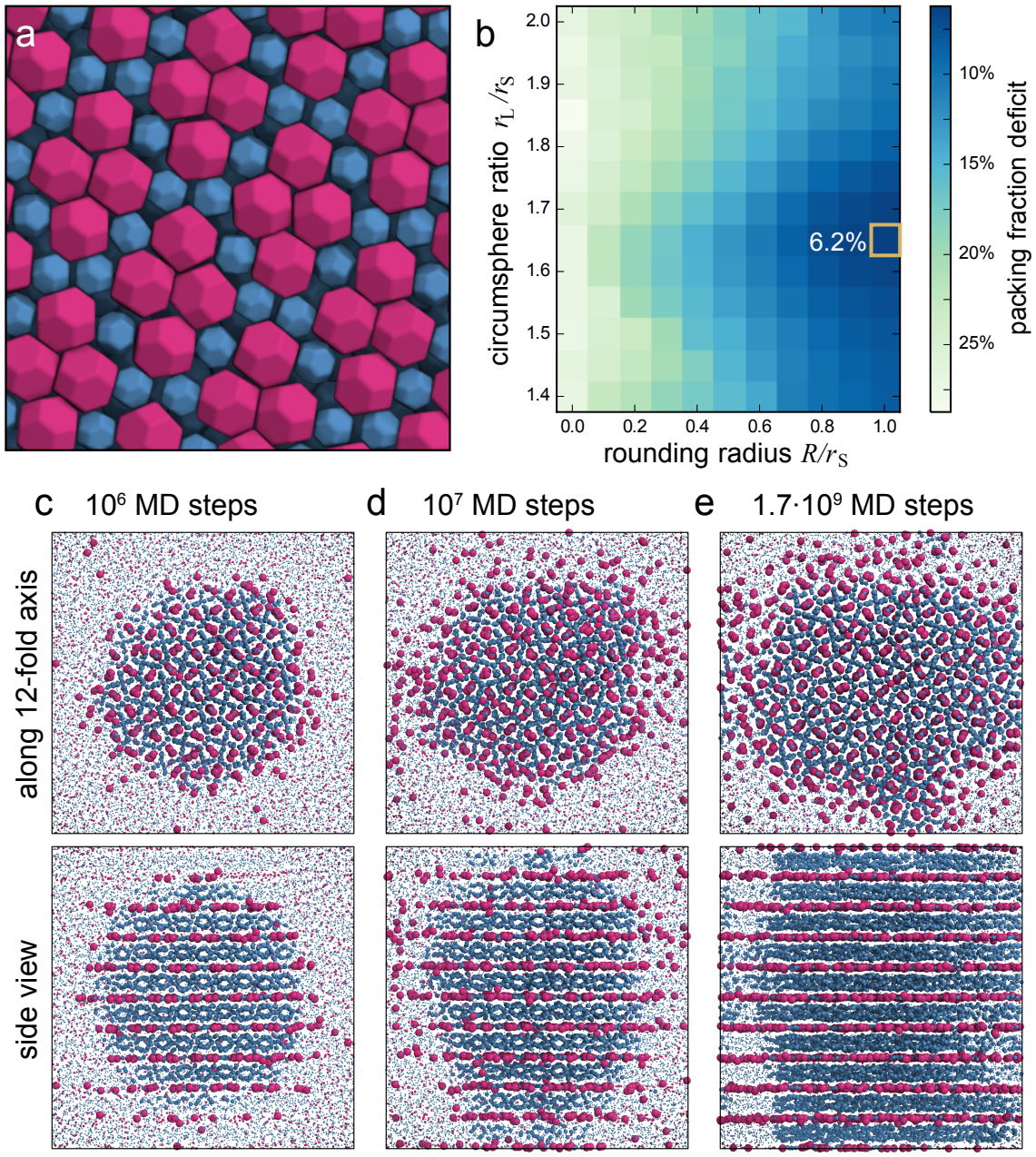


Figure 7.11: Stabilization and growth of the binary dodecagonal quasicrystal. (a) Spherotruncated octahedra as found in the $CoFe_2O_4$-Fe_3O_4 system are positioned on the decoration model for the σ-phase and densest packings numerically obtained. The particles show well-defined but not unique orientations. (b) Particle size ratio and rounding radius were varied in an attempt to discover good particle geometries that entropically favor the formation of the quasicrystalline BNSL. The optimization suggests an optimal size ratios between 1.6 and 1.7 for hard particles. (c-e) We observe the growth of the dodecagonal quasicrystal in molecular dynamics (MD) simulations of isotropic particles with A–B attraction from a seed in a simple soft particle model of size ratio 1.8. Particles are identified as solid and shown as larger spheres in the figure if they move slowly relative to their neighbors.

7.6 Insights from computer simulations

7.6.1 Hard particle model

To better understand the stabilization of the quasicrystal and to study its formation in a computer simulation, we consider its feasibility due solely to geometric effects as a binary packing of hard particles. As NC shape we choose a spheropolyhedron interpolating between a sphere and a truncated octahedron (Fig. 7.11a). Packing considerations had been successful in describing the ordering of binary sphere [133] and rod-sphere [134] mixtures. Here, however, as quantified by the packing fraction deficit (Fig. 7.11b), we find that the σ-phase always has lower density than separated single-species phases, independent of the choice of circumsphere (size) ratio or rounding radius. Packing fraction deficit is defined as the difference of the packing fraction of the densest lattice packing of the individual particles (phase separated state) and the packing fraction of the densest packing of the σ-phase in the limit of infinite pressure. This finding suggests that entropy alone is not sufficient to stabilize the quasicrystal at any pressure. Nevertheless, if we choose a rounding radius of 1 (binary spheres) and a circumradius ratio between 1.6 and 1.7, close to the NC size ratio used in experiment, then the packing fraction deficit is small at around 10%. We also observe that rounded truncated octahedra fit together well in our structure model of the quasicrystal (Fig. 7.11a) and observe metastability at packing fractions above 65%.

Electron micrographs indicate the large metallic nanoparticles in at least some samples have shapes somewhat between spheres and truncated octahedra. Systems were therefore parameterized in two different ways and parameter space was swept to optimize a candidate for dense packing. We performed parameter sweeps of hard

particle Monte Carlo simulations to find the shape, size, and orientations of particles that optimized packing for structures in the neighborhood of the candidate structure.

In the first parameterization, small particles were assumed to be round enough to be considered as spheres, while the larger particles were sphero-truncated octahedra. Parameters are the ratio of the circumradii of the two particles and the ratio of the rounding radius to the overall circumradius of the larger particle.

In the second parameterization, the particle model assumes that the rounding radius is determined primarily by the oleic acid ligand coating, which is further assumed to be similar on both particles. Both large and small particles are treated as sphereo-truncated octahedra with the same rounding radius. Adjustable parameters are the ratio of circumradii of the two particles and the ratio of the rounding radius to the overall circumradius of the smaller particle.

Simulation boxes of the σ-phase unit cell were repeatedly compressed to high pressure and relaxed using a Monte Carlo simulation following the procedure in [59]. Overlap checks consider particles as the Minkowski sum of a sphere and an Archimedean truncated octahedron [1,30]. We used the σ-phase unit cell as a starting configuration and search in the neighborhood of the candidate structure.

As a baseline we consider the densest packing of sphero-truncated octahedra (Figure 7.12) and calculate the maximum packing fraction η, which is obtained when the two particle species phase separate using the formula

$$\eta = \frac{1 + N_{\mathrm{r}} V_{\mathrm{r}}}{1 + N_{\mathrm{r}} V_{\mathrm{r}} \frac{\eta_{\mathrm{L}}}{\eta_{\mathrm{S}}}} \eta_{\mathrm{L}} \tag{7.7}$$

where $N_{\mathrm{r}} = \frac{N_{\mathrm{small}}}{N_{\mathrm{large}}} = 7.5$ is the stoichiometry (small-to-large) of the σ-phase, the volume ratio $V_{\mathrm{r}} = \frac{V_{\mathrm{small}}}{V_{\mathrm{large}}}$ is calculable in terms of R and size ratio, and η_{small} and η_{large} are determined analytically or numerically (Figure 7.12).

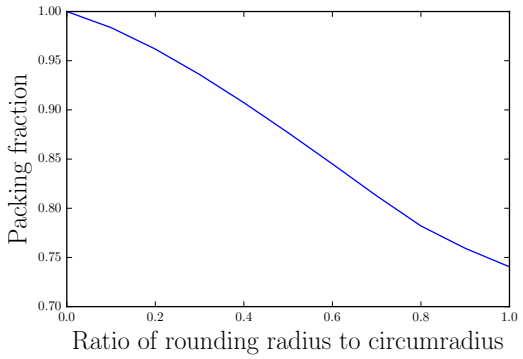


Figure 7.12: Densest obtained packing for sphero-truncated octahedra interpolating between sphere (circumradius ratio 1.0) and Archimedean truncated octahedron (circumradius 0.0). Strong agreement was found with boxes of 1, 2, 3, and 4 particles, implying that at infinite pressure the stable structure primitive unit cell contains a single particle. It is worth noting that a primitive unit cell defined by the vectors between a particle and three of its neighbors can be continuously transformed from the BCC to the FCC lattice, corresponding to the continuous transformation of the truncated octahedron to the sphere. The edge length of the primitive cell is larger than the insphere diameter of the particles during the transformation, though, as the nearest neighbors kissing faces become shifted relative to each other.

We find that while density can be readily improved over binary sphere packing, the density of the σ-phase is always lower than phase-separated dense packing (Figure 7.13). This indicates that a hard particle model alone cannot explain the σ-phase or quasicrystal at high pressures.

Nevertheless, the hard particle compressions show that truncated octahedral particles have preferred orientations. A two-fold axis of the polyhedra tends to align with the 12-fold axis of the tiling. Although the clusters of large particles give the impression of the Voronoi cells of four neighboring particles in a BCC lattice, the orientation compatible with BCC does not seem to be strongly favored. Contact tends to be between like faces. Deviation from BCC local neighborhoods may be necessary for the long-range ordered structure due to constraints on the angles between neighboring particles.

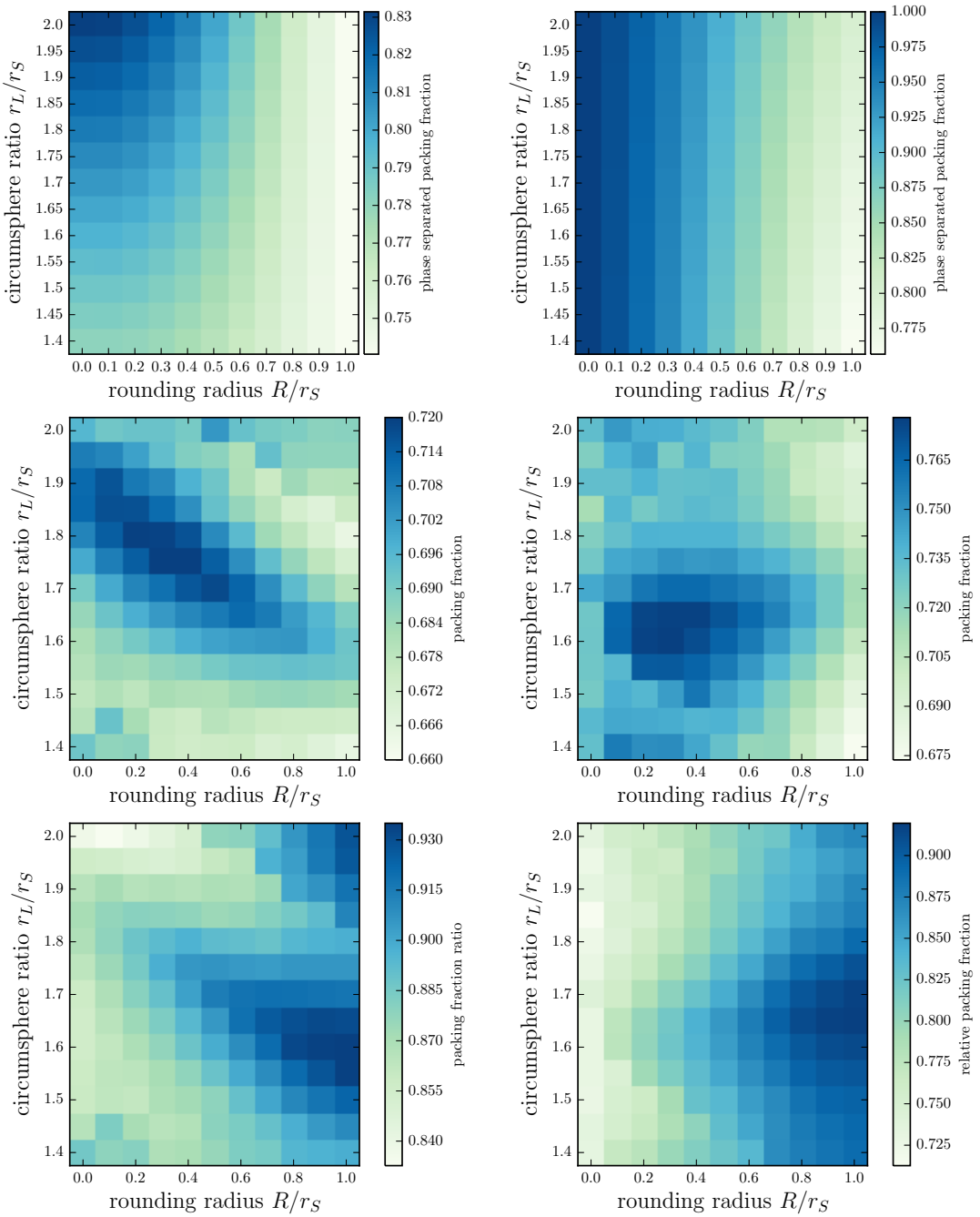


Figure 7.13: Numerically obtained maximal packing fractions. We compare mixtures of sphero-truncated octahedra and smaller spherical particles (left), and mixtures of sphero-truncated octahedra of different sizes (right), in the phase separated state (top), in the σ-phase (middle), as well as the difference of both (bottom). In all cases the stoichiometry 15:2 found in the σ-phase was used. In (bottom left), the densest packing of 0.721(3) occurred for circumsphere ratio 1.75 at rounding ratio 0.4. In (bottom right), the densest packing of 0.780(3) occurred for circumsphere ratio 1.60 at rounding ratio 0.3.

7.6.2 Shaped soft particle model

We undertook molecular dynamics simulations to try to determine what interparticle interactions were necessary to stabilize or self-assemble the observed structures. We tried to develop a particle model with an interaction potential one might reasonably assume for metallic nanoparticles with the (oleic acid) ligand coating used. The repulsive part of the potential was mapped to a WCA potential around the surface of truncated octahedra using Discrete Element Model (DEM) [135].

The softness introduced to the particle model in this way did not improve the stability of a constructed sigma phase, so we attempted to introduce an attractive element to the potential. Our DEM algorithm does not yet have an effective way to incorporate appropriate attractive forces, so an isotropic attraction was added to the potential instead. A Morse potential was parameterized to capture the shape and width of the well, but it turned out to be geometrically impossible to have a realistic attraction over a majority of the surface of the particles while maintaining the shape provided by the DEM potential.

Relaxing the parameterization of the Morse potential to accommodate a more uniform surface attraction seemed like an inappropriate complexity in an increasingly arbitrary particle model. Rather than develop a new anisotropic particle pair potential or a computationally expensive tabulated potential we decided to pursue a simpler particle model.

7.6.3 Isotropic soft particle model

We next pursued a simple isotropic binary particle model at the size ratio suggested by the packing study. To promote mixing between small (S) and large (L)

particles, we include S–L attraction via a Morse potential in addition to S–S and L–L excluded volume repulsion at the particle radius via a truncated shifted Morse potential. A fluid is seeded with several unit cells of σ-phase in a fluid of the same stoichiometry testing various values of the parameters. We observe growth of the seed for a narrow range of parameters. After initial rapid growth along the 12-fold axis, growth is significantly slower perpendicular to this axis. Despite the current inability to nucleate the quasicrystal from the melt, our results agree with the experimental findings and confirm the soundness of the structure model in Fig. 7.3a. Our simulations also establish that the width of the attractive well and in particular the particle size ratio are crucial factors in stabilizing the quasicrystal over competing phases. The best parameters for seeded growth are consistent with experimentally observed particle size ratio and surface variation. Competing phases have not been rigorously characterized.

For hard particle simulations, the best packing sphero-truncated octahedra had a circumsphere ratio of $r_L/r_S = 1.6$ and rounding radius $R/r_S = 0.3$ (Figure 7.13). We assume (per the experiments) that the hard core of the smaller particles has circumsphere diameter $d_S = 6.8$ nm, which gives $r_S = d_S/2/(1 - R/r_S) = 4.86$ nm. The ligand coating thickness is then $R = 1.46$ nm and the larger particle hard core has circumsphere diameter $d_L = 2(r_L - R) = 12.6$ nm. For the small particles, then, we choose $r_0 = 9.7$, on which the other length scales are based. We introduce an attractive Morse potential, $\Phi(r) = D(e^{-2\alpha(r-r_0)} - 2e^{-\alpha(r-r_0)})$, reminiscent of the potential from Ref. [136] to represent an orientationally averaged interparticle interaction.

The repulsive part of the potential between like particles was mapped to a shifted

and truncated Morse potential.

$$\Phi(r) = \begin{cases} D(e^{-2\alpha(r-r_0)} - 2e^{-\alpha(r-r_0)}) + D & 0 \leq r \leq r_0 \\ 0 & r_0 \leq r \end{cases}$$

Size ratio is determined by the cut-off radius r_{cut}, which is averaged to define the center of the attractive well in the S–L interaction. For the optimal size ratio of 1.8, Morse potential parameters are r_0=9.7,13.58,17.46 for S–S, S–L, and L–L particle pairs, respectively. For like particles, $r_{\text{cut}} = r_0$. For the S–L interaction, $r_{\text{cut}} = 2r_0$ and the potential and force are smoothed to zero beyond $0.9r_{\text{cut}}$. The truncation and smoothing causes a small offset of the potential minimum of about 5% of D. The best interaction width investigated is set by $\alpha = 0.3, 0.214, 0.167$ for the different types of particle pairs. The relationship between α for each pair type is determined by the ratio of r_0 of the larger to smaller particle to maintain similar potential shape for each particle type pair. Masses were assigned based on the ratio of presumed metallic core volumes. A good balance of cohesion and kinetics was found for an interaction strength of $D = 5k_{\text{B}}T$.

Molecular dynamics simulations were performed using HOOMD-blue [23,24] with a Nosé-Hoover thermostat. The Morse potential was shifted and smoothed to zero using the HOOMD-blue `xplor` option to retain differentiability. Simulations were initialized using a seeded fluid with composition equal to the known quasicrystal tile ratio

$$\frac{N_S}{N_L} = \frac{16\sqrt{3} + 7 \cdot 4}{2\sqrt{3} + 4} = 4 + 2\sqrt{3} = 7.464\ldots$$

A target density was based on the density of stable balls with open boundary conditions and later adjusted to apply a small amount of positive pressure on the seeded

system. A simulation box of 17,000 particles was initialized at about half of the target density, then thermalized and compressed for 10^5 steps. Particles were removed from the resulting configuration to make room for a ball of σ-phase, six unit cells in diameter, containing 4459 particles. Configurations with the seed in place thus ranged from 18190 to 18249 particles. The seeded fluid was first relaxed with FIRE energy minimization, then brought to the target temperature through 10^4 steps of Langevin dynamics integration. Simulations were then run for a total of up to 2×10^9 steps.

Time scale

Based on the width of the first RDF peak, it seems reasonable to choose as a basic time scale the expected time for a small particle to move about one nanometer, assumed to be roughly the distance between collisions.

In the simulations, we assume the small particles have an average kinetic energy of $(3/2)T = (1/2)mv^2$ so that

$$t = \frac{x}{\sqrt{\frac{3T}{m}}} \tag{7.8}$$

for x in simulation distance units $\mathcal{D}$, T in simulation energy units ϵ, and m in simulation mass units $\mathcal{M}$. We have chosen our simulation units such that $x = 1.0\mathcal{D} \approx 1.0$ nm. Mass $m = 112\mathcal{M}$ and $T = 1\epsilon$ gives

$$t = \frac{112}{3}\sqrt{\frac{\mathcal{M}\mathcal{D}^2}{\epsilon}} = 37.\bar{3}\tau \tag{7.9}$$

for simulation time unit τ.

For Brownian motion in the experimental system, we expect a particle to travel an average $\langle x^2 \rangle = 6Dt$ where the diffusion constant $D = \mu k_{\mathrm{B}} T$ and the Stokes mobility for a spherical particle $\mu = \frac{1}{6\pi\eta r}$.

For the solvent, hexane, we use the dynamic viscosity $\eta = 3 \times 10^{-4}$ Pa.s, or $\eta = 3 \times 10^{-4} \frac{J \cdot s}{m^3}$. Then

$$t = \frac{\langle x^2 \rangle}{6D} = \frac{\langle x^2 \rangle}{6\mu k_{\mathrm{B}} T} = \frac{\pi \eta r \langle x^2 \rangle}{k_{\mathrm{B}} T} \tag{7.10}$$

At room temperature and assuming the small particles have a radius about 4nm, then,

$$t = \frac{\pi \, 3 \times 10^{-4} \cdot 4 \times 10^{-9} \cdot 1 \times 10^{-18}}{298 \cdot 1.38 \times 10^{-23}} \text{ seconds.} \tag{7.11}$$

or about 1 nanosecond. Then $37.\bar{3}\tau = 1$ ns, or $\tau = 2.7 \times 10^{-11}$ s or about 27 picoseconds. Our simulations ran for about 1 billion τ, equivalent to about 27 milliseconds.

7.7 Conclusion

Experiment, simulation and geometric modeling is a powerful combination to find new phases of order in BNSLs. Here we have applied these techniques to a binary quasicrystal of hitherto unknown structure type. The decoration of square and triangle tiles is peculiar but similar to typical models of axial quasicrystals found in alloys [97] in the sense that it consists of well-defined layers. In contrast to atomic quasicrystals, we can perform a structure solution in real space and do not have to take a detour via the diffraction pattern. The geometric order in the BNSL tiling is comparable to atomic dodecagonal quasicrystal, suggesting that there are more similarities than differences between these two systems.

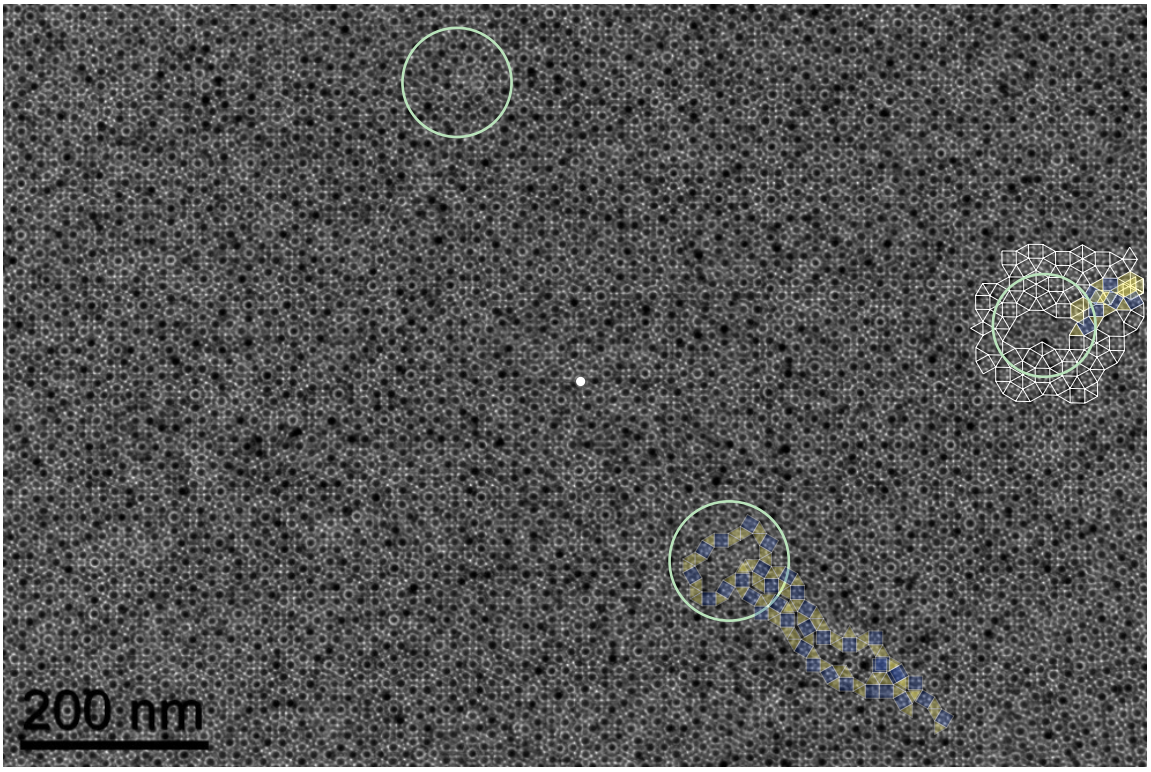


Figure 7.14: The full image of sample A with the dislocations and lattice coordinate disagreements less obtrusively annotated.

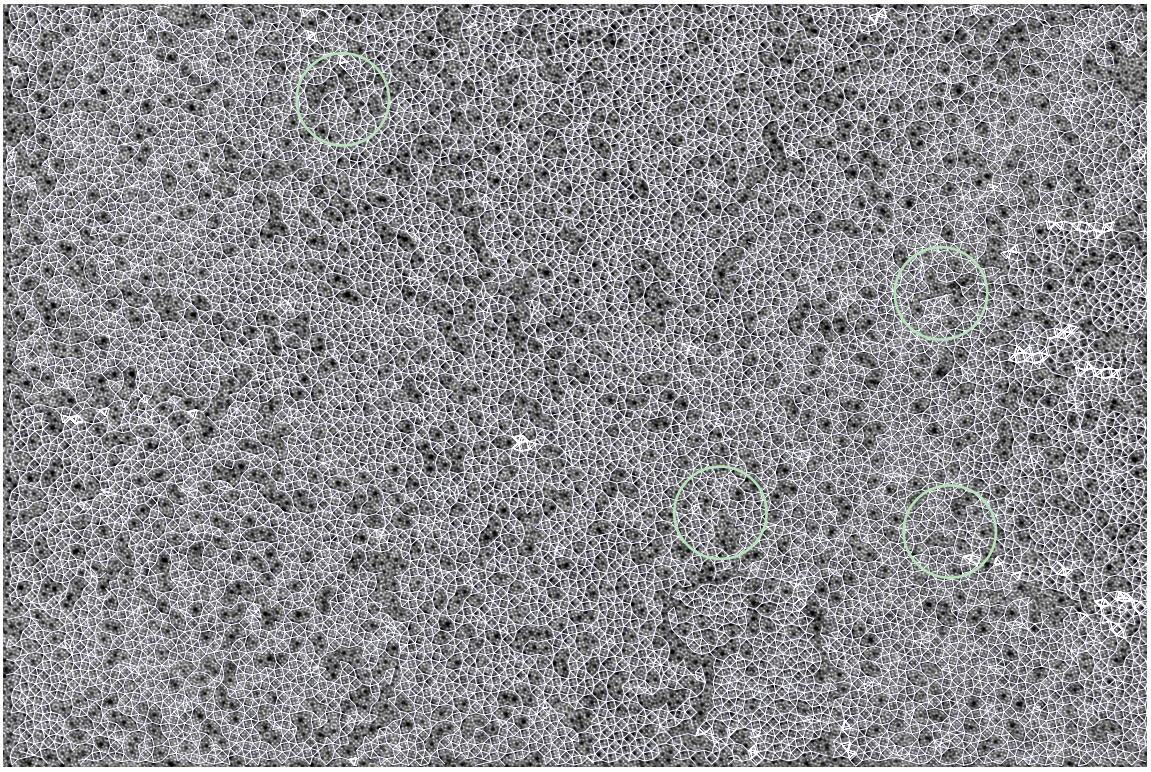


Figure 7.15: Some of the most obvious dislocations in sample B are noted with emphasis on the overlaid net.

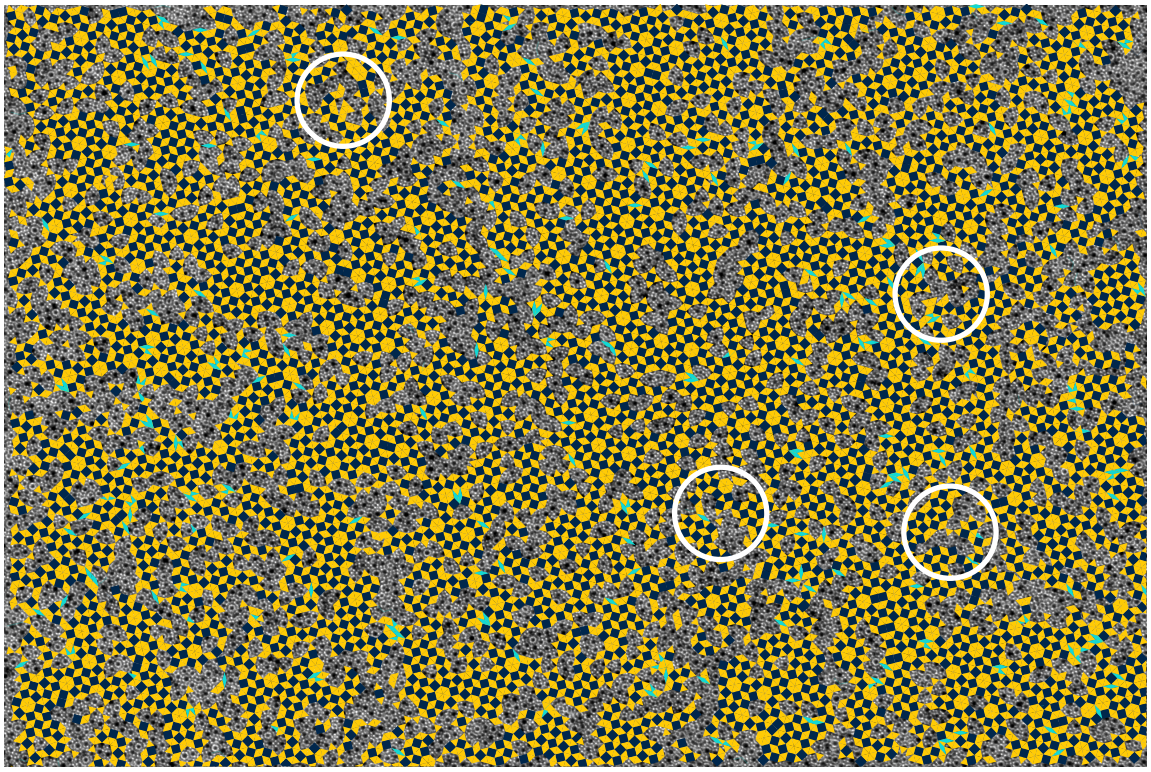


Figure 7.16: Sample B shown again square, triangle, and rhomb tiles colored. Sigma phase motifs are common.

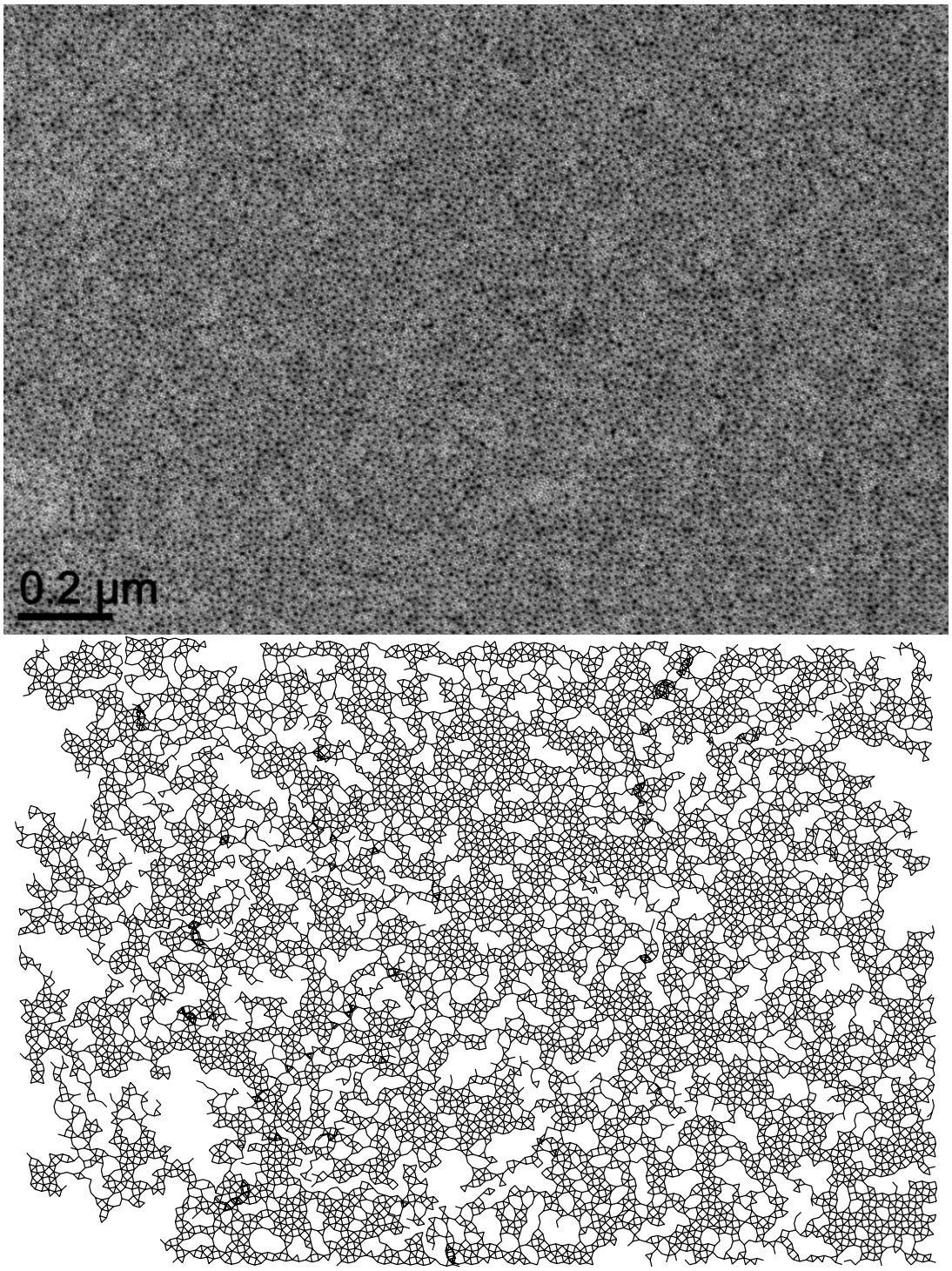


Figure 7.17: Sample C contains many defects and dislocations that can be seen in the difficulty to computationally obtain self-consistent tile coordinates in the net. Also note the occurrences of small sigma phase regions.

CHAPTER VIII

Conclusions and outlook

Through this work, we have advanced the ability for simulators and experimentalists to share thermodynamic and structural data. We have developed more powerful simulation tools as well as shown several methods and applications for measuring state functions of model colloidal (or nanoparticle) systems. In a collaboration with experimentalist colleagues, we show that a uniquely thorough and varied application of characterization and modeling techniques provides a compelling analysis of a new soft matter structure.

We have developed highly parallel thermodynamic Monte Carlo software that is finding broad application within the research group and will soon be released with the open-source HOOMD-blue particle simulation package. The software currently supports a broad variety of shapes. Careful software engineering means the code is easily extensible so we expect the functionality to continue to expand to new particle types and additional thermodynamic ensembles. Additionally, as with other HOOMD-blue tools, simulation data can be easily accessed a variety of ways for versatility in work flows or analysis techniques. As combining shape and energetic interactions continues to be a computational challenge, it seems likely that future expansions of HPMC capabilities will include finite-valued interparticle potentials.

Though developments in other parts of the HOOMD-blue code will continue to improve our ability to perform molecular dynamics with anisotropic particles, a number of discontinuous potentials (better addressed by HPMC) are historically important to advancing theoretical understanding of statistical mechanics. It is natural to incorporate and expand on these canonical particle types as HOOMD-blue becomes more powerful.

Having explored the fluid phase of a variety of hard polyhedra, it is appropriate now to apply the power and flexibility of HOOMD-blue with HPMC to the study of polyhedron solids. We should take advantage of recent developments in the understanding of shape alchemy and the ease with which new simulation and analysis code can be added. In particular, we should explore the nature of free volume in systems of greater than $3N$ degrees of freedom (anisotropic particles) and we should continue to improve our methods for determining the "best" shape to assemble a target structure. We expect thermodynamic integration for free energy estimation to continue to be an important tool, so the volume perturbation techniques explored above will benefit from refinement and optimization in more use cases.

We add image processing and a variety of geometric techniques to our tool set for the investigation of experimental data. In a long running project, we are preparing to report on an exciting new structure produced in several high-quality binary soft matter samples. The structure consists of square and triangle tiles, decorated with more than a dozen particles per tile, which we demonstrate to be arranged in a dodecagonal quasicrystalline tiling. It is clear from microscopy that the particles in some systems are regular non-spherical shapes, distinguishing them from earlier soft-matter quasicrystals, but we have not yet established the role of shape in this

system. Simulations show that the structure may be explained by allowing for an A–B particle attraction, but the mechanism of this attraction is not yet well-understood. Future work should combine interparticle attractions and shape in a search for the optimal particle parameters for self-assembly of this structure.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Joshua A. Anderson, M. Eric Irrgang, and Sharon C. Glotzer. Scalable Metropolis Monte Carlo for simulation of hard shapes. *submitted to computer physics communications*, 2015.

[2] Krste Asanovic, Ras Bodik, Bryan Christopher Catanzaro, Joseph James Gebis, Parry Husbands, Kurt Keutzer, David A Patterson, William Lester Plishker, John Shalf, Samuel Webb Williams, and Katherine A Yelick. The landscape of parallel computing research: a view from Berkeley. Technical Report UCB/EECS-2006-183, EECS Department, University of California, Berkeley, dec 2006.

[3] John Towns, Timothy Cockerill, Maytal Dahan, Ian Foster, Kelly Gaither, Andrew Grimshaw, Victor Hazlewood, Scott Lathrop, Dave Lifka, Gregory D. Peterson, Ralph Roskies, J. Ray Scott, and Nancy Wilkens-Diehr. XSEDE: Accelerating Scientific Discovery. *Computing in Science & Engineering*, 16(5):62–74, 2014.

[4] Etienne P. Bernard, Werner Krauth, and David B. Wilson. Event-chain Monte Carlo algorithms for hard-sphere systems. *Physical Review E*, 80(5):056704, nov 2009.

[5] Manon Michel, Sebastian C. Kapfer, and Werner Krauth. Generalized event-chain Monte Carlo: Constructing rejection-free global-balance algorithms from infinitesimal steps. *The Journal of Chemical Physics*, 140(5):054116, feb 2014.

[6] Grant S. Heffelfinger. Parallel atomistic simulations. *Computer Physics Communications*, 128(1-2):219–237, jun 2000.

[7] Jindal K. Shah and Edward J. Maginn. A general and efficient Monte Carlo method for sampling intramolecular degrees of freedom of branched and cyclic molecules. *The Journal of Chemical Physics*, 135(13):134121, 2011.

[8] Jason Mick, Eyad Hailat, Vincent Russo, Kamel Rushaidat, Loren Schwiebert, and Jeffrey Potoff. GPU-accelerated Gibbs ensemble Monte Carlo simulations of Lennard-Jonesium. *Computer Physics Communications*, 184(12):2662–2669, 2013.

[9] Grant S. Heffelfinger and Martin E. Lewitt. A comparison between two massively parallel algorithms for Monte Carlo computer simulation: An investigation in the grand canonical ensemble. *Journal of Computational Chemistry*, 17(2):250–265, jan 1996.

[10] Alfred Uhlherr, Stephen J. Leak, Nadia E. Adam, Per E. Nyberg, Manolis Doxastakis, Vlasis G. Mavrantzas, and Doros N. Theodorou. Large scale atomistic polymer simulations using Monte Carlo methods for parallel vector processors. *Computer Physics Communications*, 144(1):1–22, mar 2002.

[11] Ruichao Ren and G. Orkoulas. Parallel Markov chain Monte Carlo simulations. *The Journal of Chemical Physics*, 126(21):211102, 2007.

[12] C. J. O'Keeffe and G. Orkoulas. Parallel canonical Monte Carlo simulations through sequential updating of particles. *The Journal of Chemical Physics*, 130(13):134109, 2009.

[13] Joshua A. Anderson, Eric Jankowski, Thomas L. Grubb, Michael Engel, and Sharon C. Glotzer. Massively parallel Monte Carlo for many-particle simulations on GPUs. *Journal of Computational Physics*, 254(1):27–38, jul 2013.

[14] Vasilios I. Manousiouthakis and Michael W. Deem. Strict detailed balance is unnecessary in Monte Carlo simulation. *The Journal of Chemical Physics*, 110(6):2753, 1999.

[15] Tobias A. Kampmann, H. Boltz, and Jan Kierfeld. Parallelized event chain algorithm for dense hard sphere and polymer systems. *Journal of Computational Physics*, 281:864–875, jan 2015.

[16] 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):042134, apr 2013.

[17] Daan Frenkel and Berend Smit. *Understanding molecular simulation from algorithms to applications*. Academic Press, San Diego, 2 edition, 2002.

[18] Paul E. Brumby, Andrew J. Haslam, Enrique de Miguel, and George Jackson. Subtleties in the calculation of the pressure and pressure tensor of anisotropic particles from volume-perturbation methods and the apparent asymmetry of the compressive and expansive contributions. *Molecular Physics*, 109(1):169–189, jan 2011.

[19] Jens Glaser, Andrew S. Karas, and Sharon C. Glotzer. A parallel algorithm for implicit depletant simulations. *preprint*, 2015.

[20] R. Eppenga and Daan Frenkel. Monte Carlo study of the isotropic and nematic phases of infinitely thin hard platelets. *Molecular Physics*, 52(6):1303–1334, aug 1984.

[21] Eric S. Harper, Ryan Marson, Joshua A. Anderson, Greg van Anders, and Sharon C. Glotzer. Shape Allophiles Improve Entropic Assembly. *Soft Matter*, 2015.

[22] Lilian C. Hsiao, Benjamin Arthur Schultz, Jens Glaser, Michael Engel, Megan E. Szakasits, Sharon C. Glotzer, and Michael J. Solomon. Metastable orientational order of colloidal discoids. *preprint*, 2015.

[23] Joshua A. Anderson, Christian D. Lorenz, and Alex Travesset. General purpose molecular dynamics simulations fully implemented on graphics processing units. *Journal of Computational Physics*, 227(10):5342–5359, may 2008.

[24] Jens Glaser, Trung Dac Nguyen, Joshua A. Anderson, Pak Lui, Filippo Spiga, Jaime A. Millan, David C. Morse, and Sharon C. Glotzer. Strong scaling of general-purpose molecular dynamics simulations on GPUs. *Computer Physics Communications*, 192:97–107, jul 2015.

[25] HOOMD-blue. http://codeblue.umich.edu/hoomd-blue, 2015.

[26] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087, 1953.

[27] Daan Frenkel and Berend Smit. *Understanding Molecular Simulation*. Computational Science Series. Academic Press, 2 edition, 2001.

[28] Frank Smallenburg, Laura Filion, Matthieu Marechal, and Marjolein Dijkstra. Vacancy-stabilized crystalline order in hard cubes. *Proceedings of the National Academy of Sciences of the United States of America*, 109(44):17886–90, oct 2012.

[29] S. Gottschalk. Separating axis theorem. Technical report, TR96-024, Dept. of Computer Science, UNC Chapel Hill, 1996.

[30] Scott. Jacobs. *Game programming gems 7*. Charles River Media/Course Technology, Boston, MA, 2008.

[31] Wenping Wang, Jiaye Wang, and Myung-Soo Kim. An algebraic condition for the separation of two ellipsoids. *Computer Aided Geometric Design*, 18(6):531–539, jul 2001.

[32] Salvatore Alfano and Meredith L. Greer. Determining If Two Solid Ellipsoids Intersect. *Journal of Guidance, Control, and Dynamics*, 26(1):106–110, jan 2003.

[33] Carolyn L. Phillips, Joshua A. Anderson, and Sharon C. Glotzer. Pseudo-random number generation for Brownian Dynamics and Dissipative Particle Dynamics simulations on GPU devices. *Journal of Computational Physics*, 230(19):7191–7201, aug 2011.

[34] Y. Afshar, F. Schmid, A. Pishevar, and S. Worley. Exploiting seeding of random number generators for efficient domain decomposition parallelization of dissipative particle dynamics. *Computer Physics Communications*, 184(4):1119–1128, apr 2013.

[35] Tanja Schilling, Sander Pronk, Bela Mulder, and Daan Frenkel. Monte Carlo study of hard pentagons. *Physical Review E*, 71(3):036138, mar 2005.

[36] Pablo F. Damasceno, Michael Engel, and Sharon C. Glotzer. Predictive Self-Assembly of Polyhedra into Complex Structures. *Science*, 337(6093):453–457, jul 2012.

[37] Mihir R Khadilkar and Fernando A. Escobedo. Self-assembly of binary space-tessellating compounds. *Journal of Chemical Physics*, 137(19):194907, nov 2012.

[38] Michael P. Howard, Joshua A. Anderson, Arash Nikoubashman, Sharon C. Glotzer, and Athanassios Z. Panagiotopoulos. Efficient neighbor list calculation for molecular simulation of colloidal systems using graphics processing units. *preprint*, 2015.

[39] V. I. Harismiadis, J. Vorholz, and Athanassios Z. Panagiotopoulos. Efficient pressure estimation in molecular simulations without evaluating the virial. *The Journal of Chemical Physics*, 105(18):8469, nov 1996.

[40] Enrique de Miguel and George Jackson. The nature of the calculation of the pressure in molecular simulations of continuous models from volume perturbations. *The Journal of chemical physics*, 125(16):164109, oct 2006.

[41] Michael P Allen. Evaluation of pressure tensor in constant-volume simulations of hard and soft convex bodies. *The Journal of chemical physics*, 124(21):214103, jun 2006.

[42] John G. Kirkwood, Eugene K. Maun, and Berni J. Alder. Radial Distribution Functions and the Equation of State of a Fluid Composed of Rigid Spherical Molecules. *Journal of Chemical Physics*, 18(8):1040, 1950.

[43] Jean-Pierre Hansen and Ian R. McDonald. *Theory of simple liquids*. Elsevier, London; Boston, 3 edition, 2006.

[44] Jacques Vieillard-Baron. Phase Transitions of the Classical Hard-Ellipse System. *The Journal of Chemical Physics*, 56(10):4729, sep 1972.

[45] J. W. Perram, M. S. Wertheim, J. L. Lebowitz, and G. O. Williams. Monte Carlo simulation of hard spheroids. *Chemical Physics Letters*, 105(3):277–280, mar 1984.

[46] William G. Hoover. Exact Dynamical Basis for a Fluctuating Cell Model. *The Journal of Chemical Physics*, 57(3):1259, sep 1972.

[47] Robin J. Speedy. Cavities and free volume in hard-disc and hard-sphere systems. *Journal of the Chemical Society, Faraday Transactions 2*, 77(2):329, 1981.

[48] Srikanth Sastry, Thomas Truskett, Pablo Debenedetti, Salvatore Torquato, and Frank H. Stillinger. Free volume in the hard sphere liquid. *Molecular Physics*, 95(2):289–297, oct 1998.

[49] Henry Eyring and Joseph Hirschfelder. The Theory of the Liquid State. *Journal of Physical Chemistry*, 41(2):249–257, feb 1937.

[50] Zevi W. Salsburg and W. W. Wood. Equation of State of Classical Hard Spheres at High Density. *Journal of Chemical Physics*, 37(4):798, 1962.

[51] Frank H. Stillinger, Zevi W. Salsburg, and Robert L. Kornegay. Rigid Disks at High Density. *Journal of Chemical Physics*, 43(3):932, aug 1965.

[52] Zevi W. Salsburg, W. G. Rudd, and Frank H. Stillinger. Rigid Disks at High Density. II. *The Journal of Chemical Physics*, 47(11):4534, 1967.

[53] Guang-Wen Wu and Richard J. Sadus. Hard sphere compressibility factors for equation of state development. *AIChE Journal*, 51(1):309–313, jan 2005.

[54] Michael H. Huang and Po-Heng Lin. Shape-Controlled Synthesis of Polyhedral Nanocrystals and Their Facet-Dependent Properties. *Advanced Functional Materials*, 22(1):14–24, jan 2012.

[55] Joel Henzie, Michael Grünwald, Asaph Widmer-Cooper, Phillip L. Geissler, and Peidong Yang. Self-assembly of uniform polyhedral silver nanocrystals into densest packings and exotic superlattices. *Nature Materials*, 11(2):131–7, feb 2012.

[56] Kaifu Bian, Zhongwu Wang, and Tobias Hanrath. Comparing the Structural Stability of PbS Nanocrystals Assembled in fcc and bcc Superlattice Allotropes. *Journal of the American Chemical Society*, 134(26):10787–90, jul 2012.

[57] Salvatore Torquato and Y. Jiao. Dense packings of polyhedra: Platonic and Archimedean solids. *Physical Review E*, 80(4):1–21, oct 2009.

[58] Salvatore Torquato and Y. Jiao. Dense packings of the Platonic and Archimedean solids. *Nature*, 460(7257):876–9, aug 2009.

[59] Elizabeth R. Chen, Daphne Klotsa, Michael Engel, Pablo F. Damasceno, and Sharon C. Glotzer. Complexity in Surfaces of Densest Packings for Families of Polyhedra. *Physical Review X*, 4(1):011024, feb 2014.

[60] Greg van Anders, Daphne Klotsa, N. Khalid Ahmed, Michael Engel, and Sharon C. Glotzer. Understanding shape entropy through local dense packing. *Proceedings of the National Academy of Sciences*, 111(45):E4812–E4821, nov 2014.

[61] Pablo F. Damasceno, Michael Engel, and Sharon C. Glotzer. Crystalline assemblies and densest packings of a family of truncated tetrahedra and the role of directional entropic forces. *ACS Nano*, 6(1):609–14, jan 2012.

[62] Amir Haji-Akbari, Michael Engel, Aaron S. Keys, Xiaoyu Zheng, Rolfe G. Petschek, Peter Palffy-Muhoray, and Sharon C. Glotzer. Disordered, quasicrystalline and crystalline phases of densely packed tetrahedra. *Nature*, 462(7274):773–777, dec 2009.

[63] Amir Haji-Akbari, Michael Engel, and Sharon C. Glotzer. Phase Diagram of Hard Tetrahedra. *Journal of Chemical Physics*, 135(19):194101, jun 2011.

[64] Amir Haji-Akbari, Michael Engel, and Sharon C. Glotzer. Degenerate Quasicrystal of Hard Triangular Bipyramids. *Physical Review Letters*, 107(21):215702, nov 2011.

[65] Umang Agarwal and Fernando A. Escobedo. Mesophase behaviour of polyhedral particles. *Nature Materials*, 10(3):230–235, feb 2011.

[66] Ran Ni, Anjan Prasad Gantapara, Joost de Graaf, René van Roij, and Marjolein Dijkstra. Phase diagram of colloidal hard superballs: from cubes via spheres to octahedra. *Soft Matter*, 8(34), 2012.

[67] Greg van Anders, Daphne Klotsa, Andrew S. Karas, Paul M. Dodd, and Sharon C. Glotzer. Digital Alchemy for Materials Design: Colloids and Beyond. *ACS Nano*, 9(10):9542–9553, oct 2015.

[68] Yuhua Song and E. A. Mason. Equation of state for a fluid of hard convex bodies in any number of dimensions. *Physical Review A*, 41(6):3121–3124, mar 1990.

[69] Nathaniel S. Barlow, Andrew J. Schultz, S. J. Weinstein, and David A. Kofke. An asymptotically consistent approximant method with application to soft- and hard-sphere fluids. *The Journal of chemical physics*, 137(20):204102, nov 2012.

[70] Akira Isihara. Determination of Molecular Shape by Osmotic Measurement. *The Journal of Chemical Physics*, 18(11):1446, dec 1950.

[71] Gerd E. Schröder-Turk, Walter Mickel, Sebastian C. Kapfer, Fabian M. Schaller, Boris Breidenbach, Daniel Hug, and Klaus Mecke. Minkowski Tensors of Anisotropic Spatial Structure. sep 2010.

[72] Francis Birch. Finite Elastic Strain of Cubic Crystals. *Physical Review*, 71(11):809–824, jun 1947.

[73] Jean-Paul Poirier. *Introduction to the Physics of the Earth's Interior*. Cambridge University Press, 2 edition, 2000.

[74] F. D. Murnaghan. Finite Deformations of an Elastic Solid. *American Journal of Mathematics*, 59(2):235–260, 1937.

[75] F. D. Murnaghan. The Compressibility of Media Under Extreme Pressures. *Proceedings of the National Academy of Sciences of the United States of America*, 30:244–247, 1944.

[76] Pascal Vinet, John R. Smith, John Ferrante, and James H. Rose. Temperature effects on the universal equation of state of solids. *Physical Review B*, 35(4):1945–1953, 1987.

[77] Ivo Nezbeda. Virial expansion and an improved equation of state for the hard convex molecule system. *Chemical Physics Letters*, 4(1):55–58, 1976.

[78] Ivo Nezbeda, Roman Melnyk, and Andrij Trokhymchuk. A new concept for augmented van der Waals equations of state. *Journal of Supercritical Fluids*, 55(2):448–454, dec 2010.

[79] Robin J. Speedy. Statistical geometry of hard-sphere systems. *Journal of the Chemical Society, Faraday Transactions 2*, 76:693, 1980.

[80] J.L.F. Abascal and F. Bresme. Monte Carlo simulation of the equation of state of hard tetrahedral molecules. *Molecular Physics*, 76(6):1411–1421, aug 1992.

[81] Mehrdad Khanpour. Calculation of KirkwoodBuff integrals for one-component fluids using the BarkerHenderson perturbation theory. *Chemical Physics Letters*, 504(4-6):159–161, mar 2011.

[82] Andrew J. Schultz, Nathaniel S. Barlow, Vipin Chaudhary, and David a. Kofke. Mayer Sampling Monte Carlo calculation of virial coefficients on graphics processors. *Molecular Physics*, 111(4):535–543, feb 2013.

[83] Steven R. Kline. Reduction and analysis of SANS and USANS data using IGOR Pro. *Journal of Applied Crystallography*, 39(6):895–900, 2006.

[84] S. Förster, L. Apostol, and W. Bras. Scatter : software for the analysis of nano- and mesoscale small-angle scattering. *Journal of Applied Crystallography*, 43(3):639–646, apr 2010.

[85] David Babonneau. FitGISAXS : software package for modelling and analysis of GISAXS data using IGOR Pro. *Journal of Applied Crystallography*, 43(4):929–936, jul 2010.

[86] Tal Ben-Nun, Avi Ginsburg, Pablo Székely, and Uri Raviv. X+ : a comprehensive computationally accelerated structure analysis tool for solution X-ray scattering from supramolecular self-assemblies. *Journal of Applied Crystallography*, 43(6):1522–1531, oct 2010.

[87] Vincent Favre-Nicolin, Johann Coraux, Marie-Ingrid Richard, and Hubert Renevier. Fast computation of scattering maps of nanostructures using graphical processing units. *Journal of Applied Crystallography*, 44(3):635–640, apr 2011.

[88] Klaus Schmidt-Rohr. Simulation of small-angle scattering curves by numerical Fourier transformation. *Journal of Applied Crystallography*, 40(1):16–25, jan 2007.

[89] Andrew Senesi and Byeongdu Lee. Scattering functions of polyhedra. *Journal of Applied Crystallography*, 48(2):565–577, 2015.

[90] Harald Ibach and Hans Luth. *Solid-state physics: an introduction to principles of materials science*. Advanced texts in physics. Springer, Berlin ; New York, 2003.

[91] K. McInturff and P. S. Simon. The Fourier transform of linearly varying functions with polygonal support. *IEEE Transactions on Antennas and Propagation*, 39(9):1441–1443, 1991.

[92] Kevin G. Yager, Yugang Zhang, Fang Lu, and Oleg Gang. Periodic lattices of arbitrary nano-objects: modeling and applications for self-assembled systems. *Journal of Applied Crystallography*, 47(1):1–12, dec 2013.

[93] Pablo Szekely, Avi Ginsburg, Tal Ben-Nun, and Uri Raviv. Solution X-ray Scattering Form Factors of Supramolecular Self-Assembled Structures. *Langmuir*, 26(16):13110–13129, aug 2010.

[94] Matthew R Jones, Robert J. Macfarlane, Byeongdu Lee, Jian Zhang, Kaylie L. Young, Andrew J. Senesi, and Chad A. Mirkin. DNA-nanoparticle superlattices formed from anisotropic building blocks. *Nature materials*, 9(11):913–7, nov 2010.

[95] C. W. Dwiggins and D. A. Park. Calculation of the Intensity of Secondary Scattering of X-rays by Non-crystalline Materials. *Acta Crystallographica Section A*, 27:264–272, 1971.

[96] D Shechtman, I Blech, D. Gratias, and J. Cahn. Metallic phase with long-range orientational order and no translational symmetry. *Physical Review Letters*, 53(20):1951–1953, nov 1984.

[97] Walter Steurer. Twenty years of structure research on quasicrystals. Part I. Pentagonal, octagonal, decagonal and dodecagonal quasicrystals. *Zeitschrift für Kristallographie*, 219(7-2004):391–446, jul 2004.

[98] An-Pang Tsai. Discovery of stable icosahedral quasicrystals: progress in understanding structure and properties. *Chemical Society Reviews*, 42(12):5352–65, jun 2013.

[99] Xiangbing Zeng, Goran Ungar, Yongsong Liu, Virgil Percec, Andrés E. Dulcey, and Jamie K. Hobbs. Supramolecular dendritic liquid quasicrystals. *Nature*, 428(6979):157–160, mar 2004.

[100] A. Takano, W. Kawashima, A. Noro, Y. Isono, N. Tanaka, Tomonari Dotera, and Y. Matsushita. A mesoscopic Archimedean tiling having a new complexity in an ABC star polymer. *Journal of Polymer Science Part B: Polymer Physics*, 43(18):2427–2432, sep 2005.

[101] Dmitri V. Talapin, Elena V. Shevchenko, Maryna I. Bodnarchuk, Xingchen Ye, Jun Chen, and Christopher B. Murray. Quasicrystalline order in self-assembled binary nanoparticle superlattices. *Nature*, 461(7266):964–967, oct 2009.

[102] Steffen Fischer, Alexander Exner, Kathrin Zielske, Jan Perlich, Sofia Deloudi, Walter Steurer, Peter Lindner, and Stephan Förster. Colloidal quasicrystals with 12-fold and 18-fold diffraction symmetry. *Proceedings of the National Academy of Sciences of the United States of America*, 108(5):1810–1814, 2011.

[103] Jingwen Zhang and Frank S. Bates. Dodecagonal quasicrystalline morphology in a poly(styrene-b-isoprene-b-styrene-b-ethylene oxide) tetrablock terpolymer. *Journal of the American Chemical Society*, 134(18):7636–7639, may 2012.

[104] Changhong Xiao, Nobuhisa Fujita, Keiichi Miyasaka, Yasuhiro Sakamoto, and Osamu Terasaki. Dodecagonal tiling in mesoporous silica. *Nature*, 487(7407):349–353, jul 2012.

[105] Stefan Förster, Klaus Meinel, René Hammer, Martin Trautmann, and Wolf Widdra. Quasicrystalline structure formation in a classical crystalline thin-film system. *Nature*, 502(7470):215–218, oct 2013.

[106] Natalie A. Wasio, Rebecca C. Quardokus, Ryan P. Forrest, Craig S. Lent, Steven A. Corcelli, John A. Christie, Kenneth W. Henderson, and S. Alex Kandel. Self-assembly of hydrogen-bonded two-dimensional quasicrystals. *Nature*, 507(7490):86–89, mar 2014.

[107] Michael Engel and Hans-Rainer Trebin. Self-assembly of monatomic complex crystals and quasicrystals with a double-well interaction potential. *Physical Review Letters*, 98(22):225505, jun 2007.

[108] Ron Lifshitz and Haim Diamant. Soft quasicrystalsWhy are they stable? *Philosophical Magazine*, 87(18-21):3021–3030, jun 2007.

[109] Tomonari Dotera, T. Oshiro, and P. Ziherl. Mosaic two-lengthscale quasicrystals. *Nature*, 506(7487):208–11, feb 2014.

[110] Marjolein N. Van Der Linden, Jonathan P. K. Doye, and Ard A. Louis. Formation of dodecagonal quasicrystals in two-dimensional systems of patchy particles. *Journal of Chemical Physics*, 136(2012):054904, 2012.

[111] Jayasri Dontabhaktuni, Miha Ravnik, and S. Zumer. Quasicrystalline tilings with nematic colloidal platelets. *Proceedings of the National Academy of Sciences*, 111(7):2464–2469, feb 2014.

[112] Sejoon Lim, Marek Mihalkovič, and Christopher L. Henley. Matching rules from AlCo potentials in an almost realistic model. *Zeitschrift für Kristallographie*, 223(11-12):843–846, 2008.

[113] Christopher L. Henley. Random Tiling Models. In D P DiVincenzo and Paul J Steinhardt, editors, *Quasicrystals: The State of the Art*, chapter Random Til, pages 429—-524. World Scientific, Singapore ; River Edge, 1991.

[114] Keisuke Nagao, Tomoaki Inuzuka, Kazue Nishimoto, and Keiichi Edagawa. Experimental Observation of Quasicrystal Growth. *Physical Review Letters*, 115(7):075501, aug 2015.

[115] Mark Oxborrow and Christopher L. Henley. Random square-triangle tilings: A model for twelvefold-symmetric quasicrystals. *Physical Review B*, 48(10):6966–6998, sep 1993.

[116] R Collins. Statistics of a simplified two-dimensional Bemal liquid. *Proceedings of the Physical Society*, 83:553, 1964.

[117] Yves Lansac, Matthew Glaser, and Noel Clark. Discrete elastic model for two-dimensional melting. *Physical Review E*, 73(4):041501, 2006.

[118] Johannes W. Roth. Restricted square-triangle tilings. *Zeitschrift für Kristallographie International journal for structural, physical, and chemical aspects of crystalline materials*, 223(11-12):761–764, 2008.

[119] Dieter Joseph and Veit Elser. A Model of Quasicrystal Growth. *Physical Review Letters*, 79(6):1066–1069, aug 1997.

[120] Elena V. Shevchenko, Dmitri V. Talapin, Nicholas A. Kotov, Stephen O'Brien, and Christopher B. Murray. Structural diversity in binary nanoparticle superlattices. *Nature*, 439(7072):55–9, jan 2006.

[121] Zhijie Yang, Jingjing Wei, Pierre Bonville, and Marie-Paule Pileni. Beyond Entropy: Magnetic Forces Induce Formation of Quasicrystalline Structure in Binary Nanocrystal Superlattices. *Journal of the American Chemical Society*, mar 2015.

[122] Maryna I. Bodnarchuk, Elena V. Shevchenko, and Dmitri V. Talapin. Structural defects in periodic and quasicrystalline binary nanocrystal superlattices. *Journal of the American Chemical Society*, 133(51):20837–49, 2011.

[123] Maryna I. Bodnarchuk, Rolf Erni, Frank Krumeich, and Maksym V. Kovalenko. Binary superlattices from colloidal nanocrystals and giant polyoxometalate clusters. *Nano Letters*, 13(4):1699–1705, 2013.

[124] Mark P. Boneschanscher, Wiel H. Evers, Weikai Qi, Johannes D. Meeldijk, Marjolein Dijkstra, and Daniel Vanmaekelbergh. Electron tomography resolves a novel crystal structure in a binary nanocrystal superlattice. *Nano Letters*, 13(3):1312–6, mar 2013.

[125] Liron Korkidi, Kobi Barkan, and Ron Lifshitz. Analysis of Dislocations in Quasicrystals Composed of Self-assembled Nanoparticles. In *Aperiodic Crystals*, pages 117–124. Springer Netherlands, Dordrecht, 2013.

[126] Tomonari Dotera. personal communication. 2015.

[127] Benjamin Arthur Schultz. *Role of Shape in the Self-Assembly of Anisotropic Colloids*. PhD thesis, University of Michigan Ann Arbor, 2015.

[128] R Paredes, J. Aragón, and R. Barrio. Nonperiodic hexagonal square-triangle tilings. *Physical Review B*, 58(18):11990–11995, nov 1998.

[129] Marek Mihalkovič and Christopher L. Henley. Empirical oscillating potentials for alloys from ab initio fits and the prediction of quasicrystal-related structures in the Al-Cu-Sc system. *Physical Review B*, 85(9):092102, mar 2012.

[130] Walter Steurer and Sofia Deloudi. *Crystallography of Quasicrystals Concepts, Methods and Structures*, volume 126 of *Springer Series in Materials Science*. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2009.

[131] Tsutomu Ishimasa, Shuhei Iwami, Norihito Sakaguchi, Ryo Oota, and Marek Mihalkovič. Phason space analysis and structure modeling of 100 A-scale dodecagonal quasicrystal in Mn-based alloy. *Philosophical Magazine*, 95:3745, aug 2015.

[132] Michael Engel, Pablo F. Damasceno, Carolyn L. Phillips, and Sharon C. Glotzer. Computational self-assembly of a one-component icosahedral quasicrystal. *Nature materials*, 14(1):109–16, jan 2015.

[133] Adam B. Hopkins, Frank H. Stillinger, and Salvatore Torquato. Densest binary sphere packings. *Physical Review E*, 85(2):021130, feb 2012.

[134] Xingchen Ye, Jaime A. Millan, Michael Engel, Jun Chen, Benjamin T. Diroll, Sharon C. Glotzer, and Christopher B. Murray. Shape alloys of nanorods and nanospheres from self-assembly. *Nano Letters*, 13(10):4980–4988, 2013.

[135] Matthew Spellings, Ryan Marson, Joshua A. Anderson, and Sharon C. Glotzer. GPU accelerated Discrete Element Method (DEM) Molecular Dynamics. *In preparation*, 2016.

[136] Asaph Widmer-Cooper and Phillip L. Geissler. Orientational ordering of passivating ligands on CdS nanorods in solution generates strong rod-rod interactions. *Nano Letters*, 14(1):57–65, 2014.