

SOLVING ATOMIC STRUCTURE USING STATISTICAL MECHANICAL SEARCHES ON  
X-RAY SCATTERING DERIVED POTENTIAL ENERGY SURFACES

by

Christopher James Wright

Bachelor of Science  
Brown University 2014

---

Submitted in Partial Fulfillment of the Requirements

for the Degree of Masters of Science in

Chemical Engineering

College of Engineering and Computing

University of South Carolina

2016

Accepted by:

Xiao-Dong Zhou, Major Professor

Thomas Vogt, Committee Member

Mark Uline, Committee Member

Jochen Lauterbach, Committee Member

Lacy Ford, Vice Provost and Dean of Graduate Studies

© Copyright by Christopher James Wright, 2016  
All Rights Reserved.

## DEDICATION

## ACKNOWLEDGMENTS

## ABSTRACT

## TABLE OF CONTENTS

DEDICATION . . . . .	iii
ACKNOWLEDGMENTS . . . . .	iv
ABSTRACT . . . . .	v
LIST OF TABLES . . . . .	ix
LIST OF FIGURES . . . . .	x
CHAPTER 1 ATOMIC STRUCTURE: EXTRACTION AND APPLICATION . . .	2
1.1 Atomistic Goals . . . . .	2
1.2 Atomistic Experiments . . . . .	3
1.3 Atomistic Simulations . . . . .	3
CHAPTER 2 STATISTICAL MECHANICAL ENSEMBLES AND POTENTIAL ENERGY SURFACES . . . . .	4
2.1 Introduction . . . . .	4
2.2 Potential Energy Surfaces . . . . .	4
2.3 Ensembles . . . . .	6
CHAPTER 3 ATOMIC PAIR DISTRIBUTION FUNCTION: THEORY AND COMPUTATION . . . . .	11
3.1 Theory . . . . .	11

3.2 Computation . . . . .	11
3.3 Experiment . . . . .	14
3.4 Data Processing Workflow . . . . .	14
CHAPTER 4 BENCHMARKING . . . . .	26
4.1 PDF . . . . .	26
4.2 PDF with ADPs . . . . .	26
CHAPTER 5 ANNEALING AND AGGREGATION OF 2NM AU NANOPARTICLES . . . . .	27
5.1 Experiments . . . . .	27
5.2 Data Processing . . . . .	27
5.3 Data Analysis . . . . .	27
5.4 Simulation . . . . .	27
5.5 Structural Analysis . . . . .	27
5.6 Conclusions . . . . .	27
CHAPTER 6 PHASE CHANGES AND ANNEALING DYNAMICS OF $\text{Pr}_2\text{NiO}_4$ AND ITS DERIVATIVES . . . . .	28
6.1 Experiments . . . . .	28
6.2 Data Processing . . . . .	28
6.3 Data Analysis . . . . .	28
6.4 Simulation . . . . .	28
6.5 Structural Analysis . . . . .	28
6.6 Conclusions . . . . .	28

CHAPTER 7 CONCLUSION . . . . .	29
--------------------------------	----

## LIST OF TABLES

## LIST OF FIGURES

Figure 3.1 Database Loading Workflow. Data is loaded from various sources, including images and text files, into the FileStore and Meta-dataStore databases. Data is then retrieved from the databases using the databroker. . . . .	15
Figure 3.2 $Q$ resolution as a function of $Q$ . . . . .	16
Figure 3.3 Generated dead/hot pixel masks for a detector with 100 bad pixels. a) the poorly binned mask and b) the properly binned mask	18
Figure 3.4 Generated dead/hot pixel masks for a detector with 300 bad pixels. a) the poorly binned mask and b) the properly binned mask	19
Figure 3.5 Generated dead/hot pixel masks for a detector with 500 bad pixels. a) the poorly binned mask and b) the properly binned mask	19
Figure 3.6 Generated dead/hot pixel masks for a detector with 1000 bad pixels. a) the poorly binned mask and b) the properly binned mask	20
Figure 3.7 Generated beamstop holder masks for a beamstop holder with 10% transmittance. a) the raw image, b) the masked image, c) and the missed pixels . . . . .	20
Figure 3.8 Generated beamstop holder masks for a beamstop holder with 30% transmittance. a) the raw image, b) the masked image, c) and the missed pixels . . . . .	20
Figure 3.9 Generated beamstop holder masks for a beamstop holder with 50% transmittance. a) the raw image, b) the masked image, c) and the missed pixels . . . . .	21
Figure 3.10 Generated beamstop holder masks for a beamstop holder with 90% transmittance. a) the raw image, b) the masked image, c) and the missed pixels . . . . .	21
Figure 3.11 Generated beamstop holder masks which is rotated away from verticle . . . . .	21



<sup>1</sup>

## INTRODUCTION

<sup>2</sup> This is the introduction to the thesis.

3

# CHAPTER 1

## 4 ATOMIC STRUCTURE: EXTRACTION AND APPLICATION

### 5 1.1 ATOMISTIC GOALS

6 The only way to truly understand the fundamental souce of material and chemical  
7 properties is through atomic structure. The goal of atomistic engineering is to pro-  
8 duce novel structures and combinations of structures to engender new properties and  
9 functions. This includes producing stronger materials, more durable catalysts, more  
10 energy dense batteries, and many more engineering applicatons. The true power of  
11 atomistic engineering has been shown in biochemistry and pharmicuttical design. Al-  
12 though the production of drugs and biomeical treatments is usually considered to be  
13 rather far from the field of catalyst design and materials science, the atomistic nature  
14 of these fields can not be denied. The field of protiene structural analysis stands  
15 as an example of structural science, elcuidating the three dimensional coordinates of  
16 thousands of atoms. These structures are then used to describe how the molecular  
17 machinery of the biological world works, enabling the development of new drugs and  
18 treatments for diseases and a deeper understanding of how we evolved. The develop-  
19 ment of protene inhibitor drugs, which are important to so many treatments, would  
20 have not been possible without very detailed atomic structures. The asperation of  
21 this work is to create this level of accuracy and utility, generating structures which  
22 allow for the understanding of how materials work on a fundamental level.

23 1.2 ATOMISTIC EXPERIMENTS

24 Single Crystal Diffraction

25 Electron Microscopy

26 X-ray Total Scattering

27 1.3 ATOMISTIC SIMULATIONS

28 The goals of atomistic simulations are usually to produce atomic structures from  
29 quantum mechanical first principles, as in the case of Density Functional Theory  
30 (DFT), or classical approximations to quantum mechanics.

31 Density Functional Theory

32 Classical Force Field

33 Monte Carlo and Statistical Mechanics

34 Maybe put the ensemble and PES work here, since it is more general than the PDF  
35 per say. Also the rational for the gradients and fast computation make much more  
36 sense knowing we are going to be very sample happy and follow the gradient of the  
37 PES.

38

## CHAPTER 2

39

# STATISTICAL MECHANICAL ENSEMBLES AND POTENTIAL ENERGY SURFACES

41 2.1 INTRODUCTION

42 The approach taken in this work for solving the atomic structures of materials is  
43 one of optimization. The positional variables of the system are optimized so as to  
44 minimize the value of a potential energy surface (PES). The

45 2.2 POTENTIAL ENERGY SURFACES

46 A PES simply describes the potential energy of the system as a function of all its  
47 relevant coordinates in phase space, essentially providing a mapping  $\mathbb{R}^n \rightarrow \mathbb{R}$ . Usually  
48 these coordinates are the positions of the atoms  $q$  and their conjugate momenta  $p$ .  
49 Note that there could be more variables associated with the system, for instance the  
50 magnetic moments of the atoms could play a role in describing the system. In this  
51 magnetic system there would be positional variables for the atomwise spin vectors  
52 and their "momenta". Application of the term "momenta" might seem odd here, as  
53 the magnetic spin does not have a mass or a velocity. However, since the magnetic  
54 "position" is defined on the PES we need to describe its conjugate variable to properly  
55 formulate Hamiltonian dynamics and the kinetic portion of the PES.

56 **Experimentally Derived Potential Energy Surfaces**

57 Generally PESs are obtained from purely computational experiments including: ab-  
 58 initio DFT, classical approximations via the embedded atom method, or even param-  
 59 eter driven models with experimentally fitted parameters. However, one can derive  
 60 a PES from an experiment which describes how well the model reproduces the ex-  
 61 perimental data. In this case one needs a theoretical and computational framework  
 62 mapping the atomistic variables of the simulation to the same space of the data ob-  
 63 tained from the experiment. This allows the experiment to be compared directly  
 64 against the predicted data via an experimentally derived PES.

65 **Potentials**

66 For an experiment which produces 1D data, like powder diffraction, EXAFS or XPS,  
 67 the implemented potentials are:

$$\chi^2 = \sum_{a=a_{\min}}^{a_{\max}} (A_{\text{obs}} - \alpha A_{\text{calc}})^2 \quad (2.1)$$

$$Rw = \sqrt{\frac{\sum_{a=a_{\min}}^{a_{\max}} (A_{\text{obs}} - \alpha A_{\text{calc}})^2}{\sum_{a=a_{\min}}^{a_{\max}} A_{\text{obs}}^2}} \quad (2.2)$$

$$\chi_{\text{INVERT}}^2 = \frac{1}{N} \sum_j \sum_r [A_{\text{obs}}(r) - \alpha A_{j,\text{calc}}(r)]^2 \quad (2.3)$$

$$\alpha = \frac{\sum_{a=a_{\min}}^{a_{\max}} A_{\text{obs}} A_{\text{calc}}}{\sum_{a=a_{\min}}^{a_{\max}} A_{\text{calc}}^2} = \frac{\vec{A}_{\text{obs}} \cdot \vec{A}_{\text{calc}}}{|\vec{A}_{\text{calc}}|^2} \quad (2.4)$$

71 where  $A_{\text{calc}}$  and  $A_{\text{obs}}$  are the calculated and observed 1D experimental data and  $A_{\text{calc},j}$   
 72 is the calculated data for a single atom interacting with the other atoms of the system.  
 73 Note that  $A_{\text{calc}}$  has a dependence on  $q$ , the positions of the system.

74 **Forces**

$$\vec{\nabla} \chi^2 = -2 \sum_{a=a_{\min}}^{a_{\max}} (\alpha \frac{\partial A_{\text{calc}}}{\partial \gamma_{i,w}} + A_{\text{calc}} \frac{\partial \alpha}{\partial \gamma_{i,w}})(A_{\text{obs}} - \alpha A_{\text{calc}}) \quad (2.5)$$

75

$$\vec{\nabla}Rw = \frac{Rw}{\chi^2} \sum_{a=a_{\min}}^{a_{\max}} (\alpha \frac{\partial A_{\text{calc}}}{\partial \gamma_{i,w}} + A_{\text{calc}} \frac{\partial \alpha}{\partial \gamma_{i,w}})(\alpha A_{\text{calc}} - (A_{\text{obs}})) \quad (2.6)$$

76

$$\frac{\partial \alpha}{\partial \gamma_{i,w}} = \frac{(\sum_{a=a_{\min}}^{a_{\max}} A_{\text{obs}} \frac{\partial A_{\text{calc}}}{\partial \gamma_{i,w}} - 2\alpha \sum_{a=a_{\min}}^{a_{\max}} A_{\text{calc}} \frac{\partial A_{\text{calc}}}{\partial \gamma_{i,w}})}{\sum_{a=a_{\min}}^{a_{\max}} A_{\text{calc}}^2} \quad (2.7)$$

77

$$\vec{\nabla}\chi^2_{\text{INVERT}} = \frac{-2}{N} \sum_{a=a_{\min}}^{a_{\max}} \sum_j (\alpha \frac{\partial A_{j,\text{calc}}}{\partial \gamma_{i,w}} + A_{j,\text{calc}} \frac{\partial \alpha}{\partial \gamma_{i,w}})(A_{\text{obs}} - \alpha A_{j,\text{calc}}) \quad (2.8)$$

78 where  $\gamma_{i,w}$  is the  $i$ th arbitrary positional variable in the  $w$ th direction. The concept  
 79 of an "arbitrary positional variable" might seem a bit cumbersome but it allows us  
 80 to define the forces for any atomic parameter which can be represented as a vector  
 81 in 3-space. This comes in handy when trying to define the forces acting on variables  
 82 like anisotropic displacement parameters or atomic magnetic spins.

83 DISCUSS INVERT A BUNCH. ALSO COMPARE RW AND CHI\*\*2, POTEN-  
 84 TIALY WITH A FIGURE.

85 2.3 ENSEMBLES

86 While PESs describe which atomic configurations are the most desirable and how  
 87 the atoms would like to get there, the ensemble describes how the atoms move on  
 88 the PES. The abstraction of the PES from the ensemble is an important one, as it  
 89 allows for the reuse and exchange of both PESs and ensembles for a wide array of  
 90 problems. Statistical mechanical ensembles can be described in two ways, analytically  
 91 and stochastically. For long simulation times and fine enough numerical or analytical  
 92 integration these two descriptions should be identical. In either case one starts by  
 93 defining the Hamiltonian  $\mathcal{H}$  as the total energy of the system. Thus, the Hamiltonian  
 94 is described as the sum of the potential  $U(q)$  and kinetic  $K(p)$  energies, where  $q$  is  
 95 the positions of the atoms and  $p$  is their momenta

$$\mathcal{H}(q, p) = U(q) + K(p) \quad (2.9)$$

96 where  $K(p) = \frac{1}{2} \sum_i \frac{p_i^2}{m_i}$  and  $i$  denotes the  $i$ th particle. Analytically one generally defines  
 97 a partition function, which describes the sum of probabilities over all potential atomic  
 98 states.

$$\Xi = \sum_i P_i(q, p)$$

99 where  $P_i$  is the probability of the  $i$ th state and is a function of the total energy of  
 100 that state. This partition function can then be used to obtain the probability of any  
 101 specific state.

## 102 Hamiltonian Monte Carlo

In order to model dynamics we need to describe the motion of the particles in our system, thus:

$$\frac{dq_i}{dt} = \frac{\partial \mathcal{H}}{\partial p_i} = p_i \quad (2.10)$$

$$\frac{dp_i}{dt} = -\frac{\partial \mathcal{H}}{\partial q_i} = -\vec{\nabla}U \quad (2.11)$$

Using these equations we can derive the position and momentum vectors at any point in time using the leap-frog algorithm:

$$p_i(t + \delta t/2) = p_i(t) - \frac{\delta t}{2} \frac{\partial}{\partial q_i} U(q(t)) \quad (2.12)$$

$$q_i(t + \delta t) = q_i(t) + \delta t * p_i(t + \delta t/2) \quad (2.13)$$

$$p_i(t + \delta t) = p_i(t + \delta t/2) - \frac{\delta t}{2} \frac{\partial}{\partial q_i} U(q(t + \delta t)) \quad (2.14)$$

103 Note that  $\frac{\partial}{\partial q_i}$  is the gradient with respect to  $q$  where  $i$  denotes the  $i$ th atom being  
 104 moved. Using this notation the gradient is

$$\vec{\nabla}U = \begin{bmatrix} \frac{\partial U}{\partial q_{0,x}} & \frac{\partial U}{\partial q_{0,y}} & \frac{\partial U}{\partial q_{0,z}} \\ \vdots & \frac{\partial U}{\partial q_{i,w}} & \vdots \\ \frac{\partial U}{\partial q_{n,x}} & \frac{\partial U}{\partial q_{n,y}} & \frac{\partial U}{\partial q_{n,z}} \end{bmatrix} = \begin{bmatrix} \vec{\mathcal{F}}_0 \\ \vdots \\ \vec{\mathcal{F}}_i \\ \vdots \\ \vec{\mathcal{F}}_n \end{bmatrix} \quad (2.15)$$

105 where  $\frac{\partial}{\partial q_{i,w}}$  is the derivative with respect to  $q$  where  $w$  denotes direction of the deriva-  
 106 tive ( $x$ ,  $y$ , or  $z$ ),  $n$  is the number of atoms and  $U$  is the potential which depends on  
 107  $q$ , and  $\vec{F}_i$  is the "force" on the  $i$ th atom.

108 **No-U-Turn-Sampling**

109 **Grand Canonical Ensemble**

110 **Ensemble description**

111 In the Grand Canonical Ensemble (GCE) two sets of variables are allowed to change,  
 112 the atomic positions and the total number of atoms and their associated identities.  
 113 These two variables are controlled by temperature and chemical potential. The par-  
 114 tition function is

$$\Xi = e^{-\beta(\mathcal{H}+\mu)} \quad (2.16)$$

115 This is translated into a Monte Carlo system, producing Grand Canonical Monte  
 116 Carlo (GCMC).

117 **Grand Canonical Monte Carlo**

118 While the probabilities for atomic motion are the same as in the Canonical Ensemble,  
 119 the addition or removal of an atom have their own probabilities. For the addition of  
 120 an atom the probability is formally:

$$\min[1, \frac{V}{(N+1)\Lambda(T)^3} e^{-\beta\Delta U + \beta\mu}] \quad (2.17)$$

121 Similarly the removal of an atom has the probability:

$$\min[1, \frac{(N)\Lambda(T)^3}{V} e^{-\beta\Delta U - \beta\mu}] \quad (2.18)$$

122 However, both of these equations depend of the overall simulation volume and the  
 123 thermal wavelength, which is undesirable as these are not really properties that we

124 are of interest to these simulations. Thus, we roll them into the definition of the  
125 chemical potential, essentially setting the base chemical potential to counteract these  
126 effects. This makes certain that our simulation does not change if we change the  
127 overall cell volume. A GCMC move consists of creating a new atomic configuration,  
128 where an atom has been added or removed, and checking the above criteria. However,  
129 previous results have shown that this method is computationally expensive in dense  
130 liquids, and exceedingly expensive in solid materials. The long simulation times  
131 are due to the random nature of the atomic additions or removals which produce:  
132 over-tightly packed atoms, atoms in the middle of nowhere, or unphysical vacancies.  
133 These configurations are rejected by the GCMC criteria but their probability of being  
134 sampled is much higher than configurations which are lower in energy, since the  
135 number of incorrect ways to add/remove atoms is much larger than the correct ways.  
136 Thus we have implemented methods for biasing the atomic addition positions and  
137 the atomic removals toward configurations which are more likely to be accepted.

### 138 GCMC biasing

139 The first method is to remove some of the excess options from the probability pool.  
140 Initially the insertion positions are calculated at random using a random number gen-  
141 erator and scaled to the size of the simulation cell. This produces probabilities which  
142 have floating point level precision, which is effectively infinite. While this produces  
143 a potentially infinite number of ways to create energetically favorable configurations,  
144 the infinite ways to produce bad configurations is much larger. Thus we can limit this  
145 by moving to voxels. In this case atoms are added to the center of voxels which have  
146 a pre-set resolution, limiting our total number of valid addition points. While this  
147 could produce some problems with ergodicity, we avoid this by allowing the atoms to  
148 translate throughout the system. Each voxel has a probability of being tried:

$$P_{i,j,k} = \frac{xyz}{abc} \quad (2.19)$$

149 where  $x, y, z$  and  $a, b, c$  are the resolutions and cell side lengths in the cardinal di-  
150 rections, respectively. While this does help to limit the total probability space it  
151 does not tell us which voxels are likely to lead to better configurations, leading to  
152 many rejected atomic additions. To combat this issue we can weigh the individual  
153 voxels, giving more probability to voxels which show promise and less to those with  
154 less likelihood to be accepted.

155 The approach most likely to yield success would be to measure the change in  
156 potential energy associated with the addition of an atom at the center of the voxel  
157 where the probability of a voxel to be tried is:

$$P_{i,j,k} = \frac{e^{\beta\Delta U_{i,j,k}}}{\sum_{i,j,k} e^{\beta\Delta U_{i,j,k}}} \quad (2.20)$$

158 where  $\Delta U_{i,j,k}$  is the change in energy. However, calculating  $\Delta U_{i,j,k}$  can be particu-  
159 larly expensive, especially when calculating scattering from atomic positions. The  
160 computational expense can be mitigated by using a cheaper potential, if only for the  
161 evaluation of the voxel energy, as previously shown. Similar to previous work we can  
162 use the Lennard Jones potential to approximate the addition potential.

163

## CHAPTER 3

164

### ATOMIC PAIR DISTRIBUTION FUNCTION:

165

### THEORY AND COMPUTATION

166 3.1 THEORY

167 To properly understand the PDF and its limitations we need to derive its mathemat-  
168 ics. The following derivation has been performed numerous times but most recently  
169 and completely by Farrow and Billinge, it is reproduced here for clarity and com-  
170 pleteness.

171 **Derivation**

172 Consider a wave incident on a volume of variable density...

173 **Analytical Gradients**

174 Many optimization algorithms and simulations methodologies, including HMC, re-  
175 quire not only the potential energy of a given configuration but also the forces acting  
176 on that configuration. These forces are described by the gradient of potential energy  
177 of the system.

178 3.2 COMPUTATION

179 Simply deriving the equations for the PDF is not enough. The many body nature of  
180 the PDF equation make analytical solution of the structure from the PDF impossible.

181 Thus, the PDF must be computed from a structural candidates and compared against  
182 experimental results to evaluate the reliability of the model.

183 **HPC and GPUs**

184 To properly solve the structure of materials the PDF will need to be computed many  
185 times and checked against experimental results. This requires computation of the  
186 PDF, potentially over many atoms. Calculating these PDFs requires a fast, highly  
187 parallelized, computational framework.

188 **GPUs and Parallelization**

189 Computing the PDF is an embarrassingly parallel problem. The basic procedure is  
190 to calculate the reduced structure factor  $F(Q)$  for each atom pair and momentum  
191 transfer vector, sum over all the atom pairs, and Fourier transform the structure to  
192 the PDF. The first part of this procedure is perfectly parallelizable, as each atom pair is  
193 separate from the others. The summation over all the atomic reduced structure factors  
194 can be parallelized via distributed summing. Lastly the FFT can be parallelized using  
195 existing parallel FFT algorithms.

196 GPUs are particularly well suited to the task of computing PDFs. GPU chip  
197 architecture is designed to perform many tasks simultaneously by having potentially  
198 thousands of cores.

199 **Map from ij space to k space**

200 The above equations, although formally correct, are very inefficient.  $F(Q)$  and its  
201 gradient are indexed over all the atoms twice, however there are symmetries that  
202 allow us to only compute over the atom pairs essentially mapping from an  $n \times n$  space,  
203  $ij$  space, to a  $\frac{n(n-1)}{2}$  space,  $k$  space. For  $F(Q)$  we apply the following mapping where  
204  $E$  denotes the atomic coordinates in  $ij$  space,  $E'$  denotes  $F(Q)$  before the summation

$$\begin{array}{ccccc}
& & \psi & & \Sigma \\
E & \xrightarrow{\quad} & E' & \xrightarrow{\quad} & Z \\
\phi \downarrow & & & & \nearrow \Sigma' \\
& & B & \xrightarrow{\quad} & B' \\
& & \psi' & &
\end{array}$$

in  $ij$  space,  $B$  denotes the atomic pairs in  $k$  space,  $B'$  denotes  $F(Q)$  in  $k$  space, and  $Z$  denotes the final summed  $F(Q)$ . For the operators,  $\phi$  denotes the mapping from  $ij$  space to  $k$  space  $k = j + i * \frac{i-1}{2}$ ,  $\psi$  and  $\psi'$  denote the  $F(Q)$  operation in  $ij$  and  $k$  space, respectively.  $\Sigma$  denotes the sum over all the atoms.

To properly define  $\Sigma'$  we must establish whether  $F(Q)$  is an even function. We can accomplish this by examining each of the portions of  $F(Q)$ ,  $\alpha, \beta, \tau, \Omega$ .  $\Omega$  is even, since  $r_{ij}$  is the interatomic distance, which is the same despite a flip of indicies,  $Q$  does not depend on the atomic indicies, and since  $Qr_{ij}$  is even so is  $\sin Qr_{ij}$ . Thus,  $\Omega$  is even. Providing similar analysis to  $\tau$  we can see that while  $\vec{u}_{ij}$  is odd, so is the unit displacement vector between the two atoms, thus the two odds cancel out. Intuitivly this makes sense, since the  $F(Q)$  equation is fundamentally interested in the interatomic distances which is even. Thus, switching atom indicies does not change  $F(Q)$ . Due to the even nature of the  $F(Q)$  operator the  $\Sigma'$  operator sums over all the atom pairs, and multiplies by two to reflect the double counting of the  $\Sigma$  operator.

For the gradient a similar mapping is used:

$$\begin{array}{ccccc}
& & \psi & & \Sigma \\
E & \xrightarrow{\quad} & E' & \xrightarrow{\quad} & Z \\
\phi \downarrow & & & & \nearrow \tilde{\phi}\Sigma' \\
& & B & \xrightarrow{\quad} & B' \\
& & \psi' & &
\end{array}$$

219

In this mapping, however, we use the  $\tilde{\phi}\Sigma$  operator. This operator simultaniously performs a reverse mapping from  $k$  to  $ij$  space, and a summation with the correct

222 symmetry. In this case the  $\psi$  and  $\psi'$  operators, which denote the  $\vec{\nabla}F(Q)$  operator  
223 in  $ij$  and  $k$  space, are antisymmetric. Intuitivly this makes sense as an extension of  
224 Newton's Second Law, since each particle's interation is felt oppositely by its partner.

## 225 Periodic Boundary Conditions

226 Periodic boundary conditions can be helpful when simulating extended solids or large  
227 nanoparticles. In this case all the non-crystallinity is contained within the simulation  
228 box and the box is repeated to create the longer distance peaks observed in the PDF.  
229 To perform this we can break up the Debye equation into two main parts, the part  
230 that describes the interatomic distances within the simulation box and those between  
231 boxes. Neglecting the thermal motion portion:

$$F(Q) = \frac{1}{N\langle f \rangle^2} \left( \sum_{j \neq i} f_i^*(Q) f_j(Q) \frac{\sin(Qr_{ij})}{r_{ij}} + \sum_{i,j} f_i^*(Q) f_j(Q) \frac{\sin(QR_{ij})}{R_{ij}} \right) \quad (3.1)$$

232 where

$$R = |\vec{r} + \vec{u}| \quad (3.2)$$

$$\vec{u} = \gamma_1 * \vec{a} + \gamma_2 * \vec{b} + \gamma_3 * \vec{c} \quad (3.3)$$

## 233 3.3 EXPERIMENT

234 PDF experiments are generally performed at synchrotron light sources, as only these  
235 sources can provide the need flux, energy, and high momentum transfer vectors needed  
236 to obtain reliable PDFs.

## 237 3.4 DATA PROCESSING WORKFLOW

238 Processing the raw pixel intensities to the PDF is very important as we are extracting  
239 most of our interesting information out of very high  $Q$  data. This data relies on good  
240 statistics and sound background subtraction. Talk about papers from Billinge Group

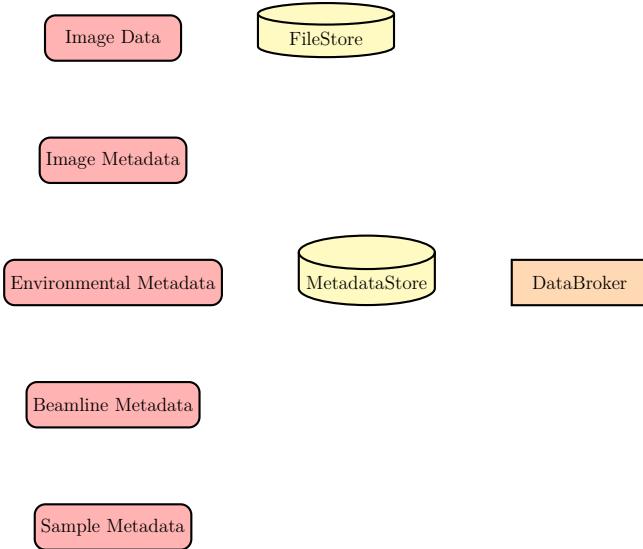


Figure 3.1: Database Loading Workflow. Data is loaded from various sources, including images and text files, into the FileStore and MetadataStore databases. Data is then retrieved from the databases using the databroker.

241 with thin film PDF and dilute NP solutions. Diagram of the overall data processing  
 242 workflow. Discuss the NSLS-II data stack.

### 243 **MetadataStore Side Loading**

244 Design of sidewinder-spec for loading the data into metadatastore. Most of the design  
 245 considerations went into the loaders, which are different for each experiment.

### 246 **Automated Image Azimuthal Integration**

247 Mux data as needed. Use pyFAI to get the radial distance array. Note that to  
 248 properly mask and integrate the system we need to compute the bin edges for the  
 249 pixels. The bin edges change as a function of  $Q$ , as the angle subtended by a pixel  
 250 shrinks essentially giving high  $Q$  pixels more resolution.

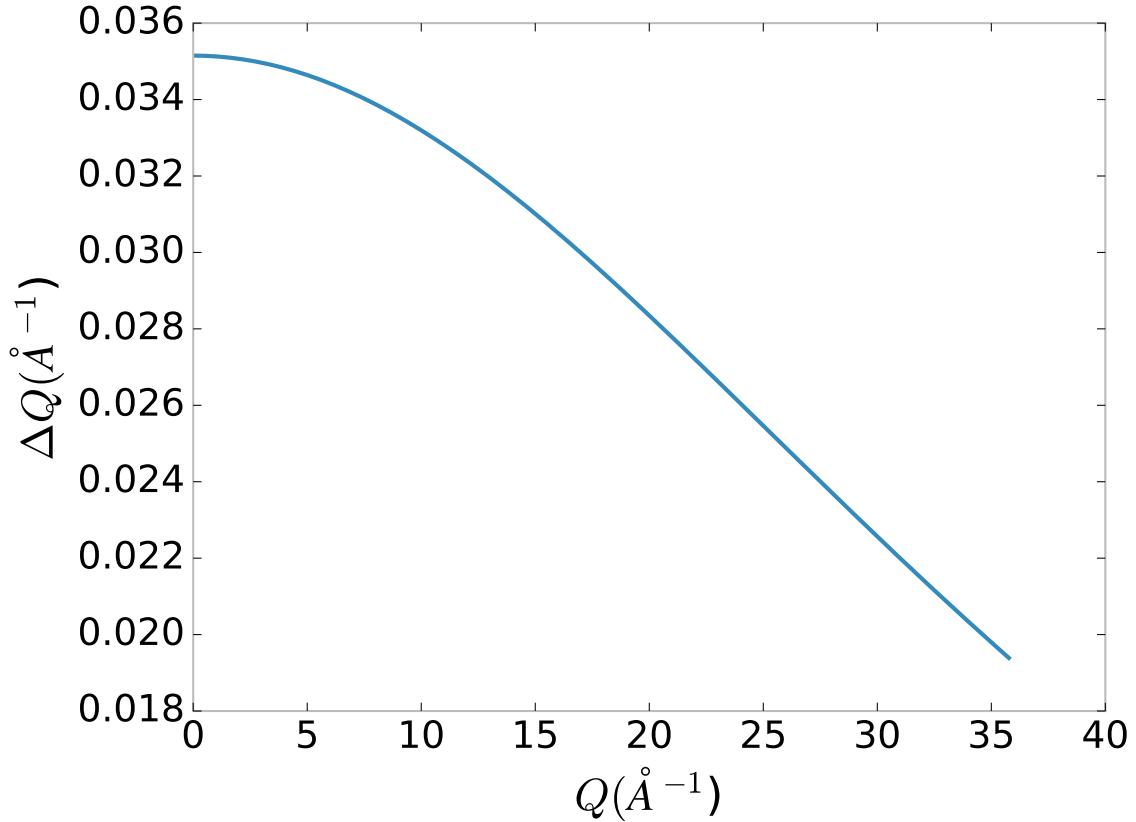


Figure 3.2:  $Q$  resolution as a function of  $Q$ .

251 **Detector  $Q$  resolution**

252 **Automated Mask Generation**

253 **Introduction**

254 Detector masking is an important part of any x-ray scattering workflow as dead/hot  
 255 pixels, streak errors, and beamstop associated features can be averaged into the data  
 256 changing the signal and its statistical significance. While some features, like the  
 257 beamstop holder, can be easily observed and masked by hand other are much more  
 258 difficult to observe even on large computer monitors. Additionally, while dead/hot  
 259 pixels and streaks are usually static the hot pixels associated with textured or sin-  
 260 gle crystal scattering or cosmic rays are not. Thus, coming up with an automated  
 261 method for finding such erroneous pixels is important, especially as high flux diffrac-

262 tion beamlines can generate data very quickly.

263 While this problem can be quite complex in the most general case, we can use the  
264 annular symmetry of the powder scattering pattern to our advantage, by comparing  
265 a pixel against pixels in the same ring. Since non-textured powder scattering should  
266 produce the same pixel intensity for a given ring we can mask any pixels which are  $\alpha$   
267 standard deviations away from the mean. This method relies on the aforementioned  
268 pixel binning algorithm, as using miss sized bins will cause some pixels which should  
269 be in separate rings to be put together, and others which should be in the same ring  
270 to be separated. In that case the masking algorithm will overestimate the number of  
271 pixels to be masked due to the additional statistical variation in the sample.

## 272 **Algorithm Design**

273 The masking algorithm procedure takes in the image and a description of the pixel  
274 positions in either distance from the point of incidence or in  $Q$ . The image is then  
275 integrated twice, producing both the mean  $I(Q)$  and the standard deviation of each  
276  $I(Q)$  ring. The mask is created by comparing the pixel values against each ring's  
277 standard deviation and threshold  $\alpha$ . Note that the threshold can be a function of  
278 distance from the point of incidence or  $Q$ .

## 279 **Test Cases**

280 To study the effectiveness of the masking we ran the algorithm against both simulated  
281 experimental data. In the case of the simulated data four systems were created: 1)  
282 dead/hot pixels with varying numbers of defective pixels, 2) beamstop holder with  
283 varying beamstop holder transmittance, 3) rotated beamstop holder with varying  
284 beamstop holder transmittance, and 4) beamstop holder with dead/hot pixels. The  
285 base scattering was produced by

$$I = 100 \cos(50r)^2 + 150 \quad (3.4)$$

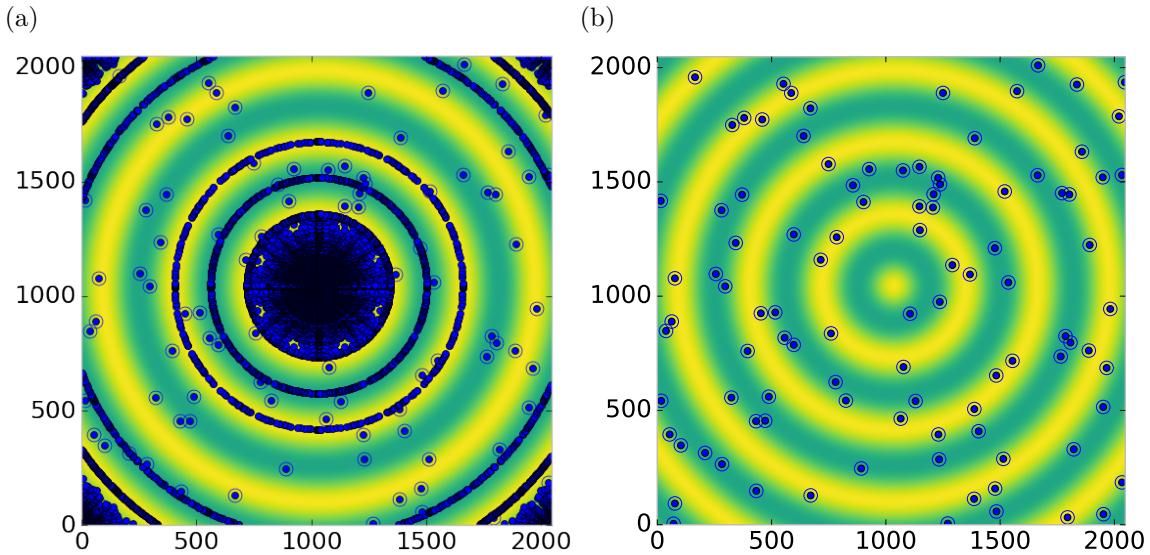


Figure 3.3: Generated dead/hot pixel masks for a detector with 100 bad pixels. a) the poorly binned mask and b) the properly binned mask

where  $r$  is a pixel's distance from the beam point of incidence. The positions of the dead/hot pixels were chosen at random as was the dead or hot nature of the defect. Dead pixels had values from 0 to 10, while hot pixels had values from 200 to 255. The beamstop was positioned at the vertical center of the detector with an initial width of 60 pixels and final width of 120 pixels. The height of the beamstop was 1024 pixels. The beamstop was calculated to attenuate the x-ray scattering signal at various transmittance, as various beamstop holder materials have different transmittance. Two versions of the masking algorithm were run for each test case, one using the standard even bin sizes for the integration step, and one where the bin sizes are tuned to the pixel  $Q$  resolution as discussed in 3.4.

## Results and Discussion

Figures 3.3-3.10 show the results of the masking algorithm on simulated images. The dead/hot pixel masking shows the importance of using the  $Q$  resolution based bin sizes as the even bin based mask have a tendency to over mask the image, removing pixels

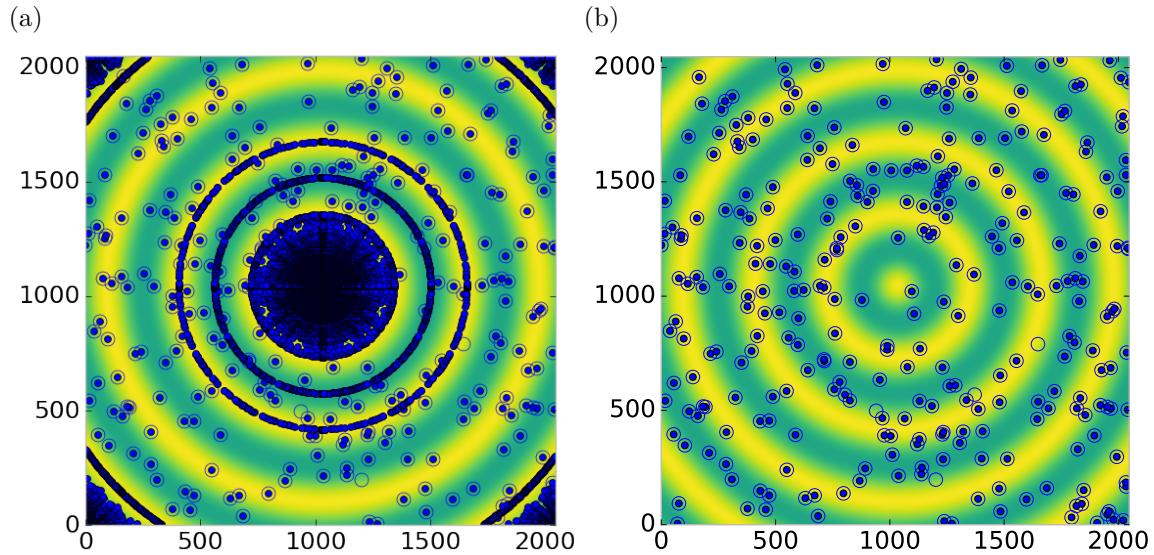


Figure 3.4: Generated dead/hot pixel masks for a detector with 300 bad pixels. a) the poorly binned mask and b) the properly binned mask

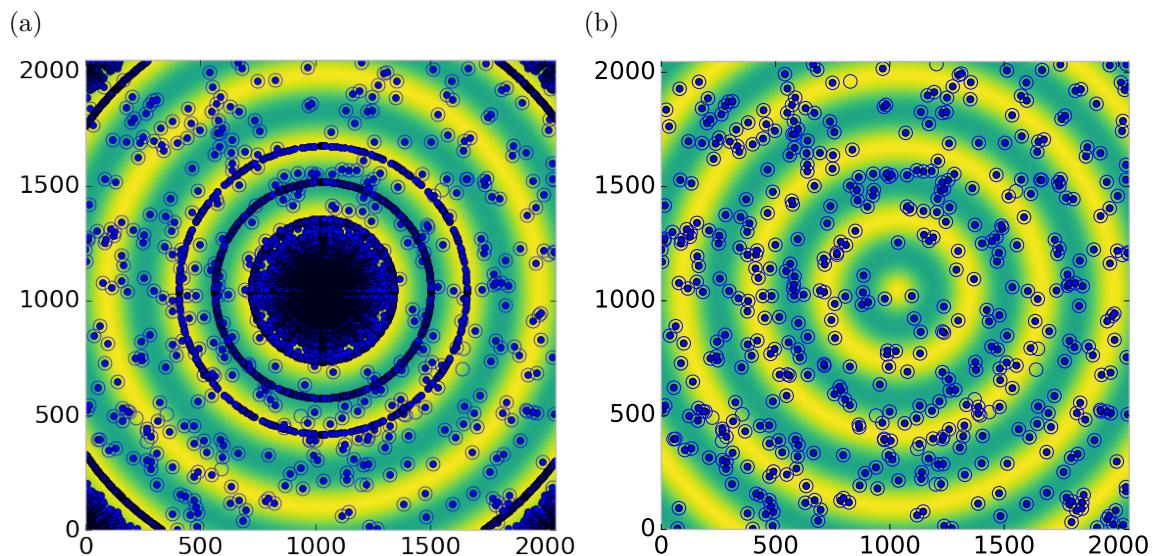


Figure 3.5: Generated dead/hot pixel masks for a detector with 500 bad pixels. a) the poorly binned mask and b) the properly binned mask

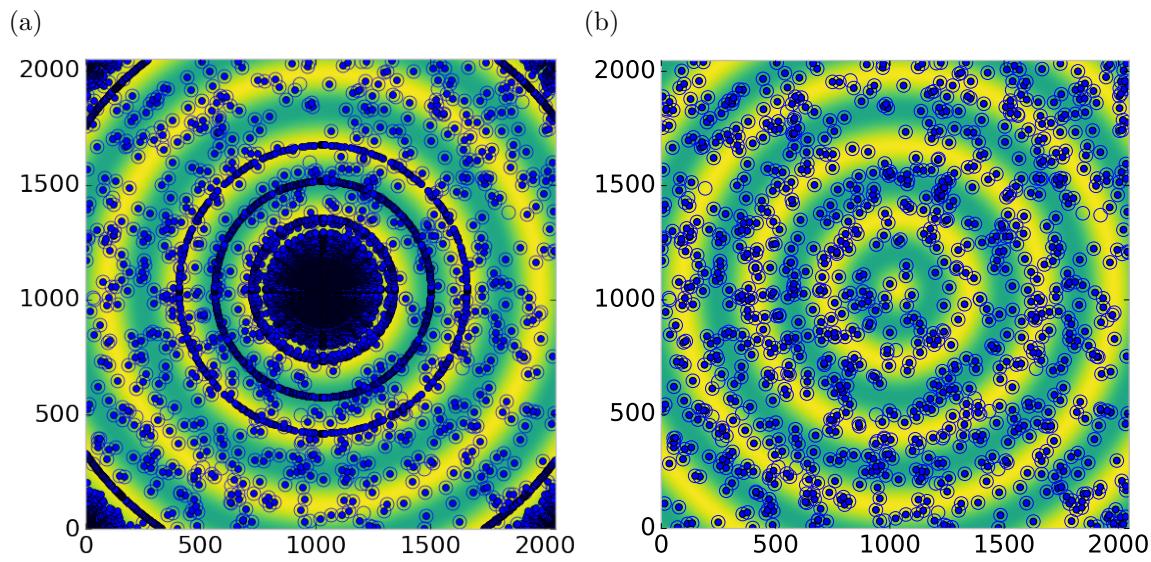


Figure 3.6: Generated dead/hot pixel masks for a detector with 1000 bad pixels. a) the poorly binned mask and b) the properly binned mask

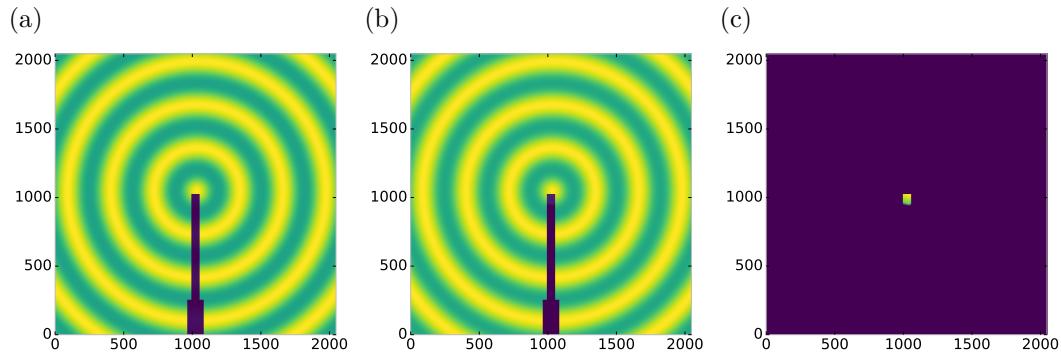


Figure 3.7: Generated beamstop holder masks for a beamstop holder with 10% transmittance. a) the raw image, b) the masked image, c) and the missed pixels

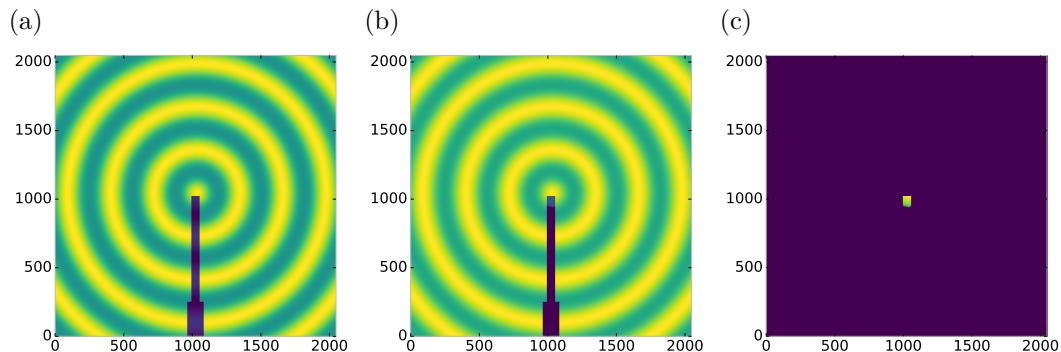


Figure 3.8: Generated beamstop holder masks for a beamstop holder with 30% transmittance. a) the raw image, b) the masked image, c) and the missed pixels

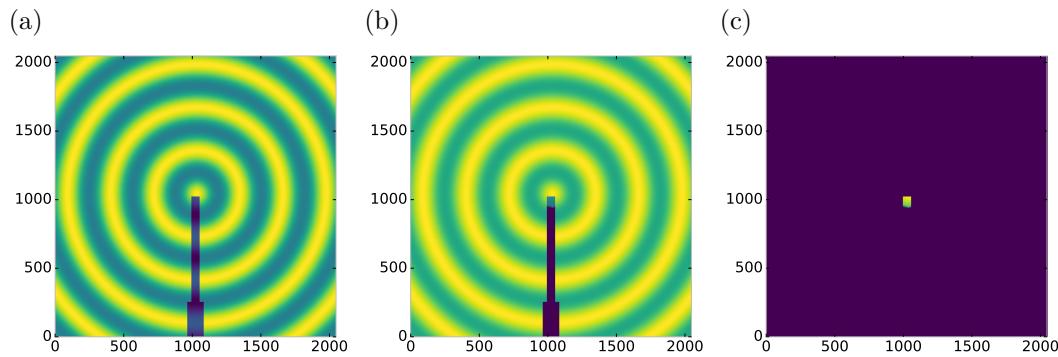


Figure 3.9: Generated beamstop holder masks for a beamstop holder with 50% transmittance. a) the raw image, b) the masked image, c) and the missed pixels

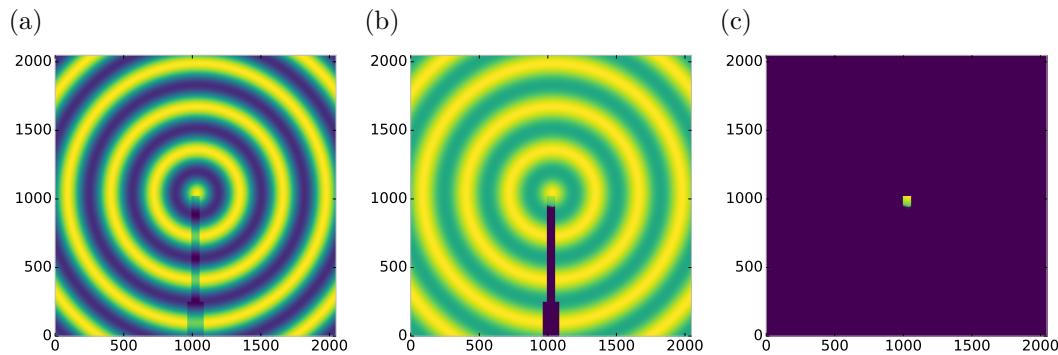


Figure 3.10: Generated beamstop holder masks for a beamstop holder with 90% transmittance. a) the raw image, b) the masked image, c) and the missed pixels

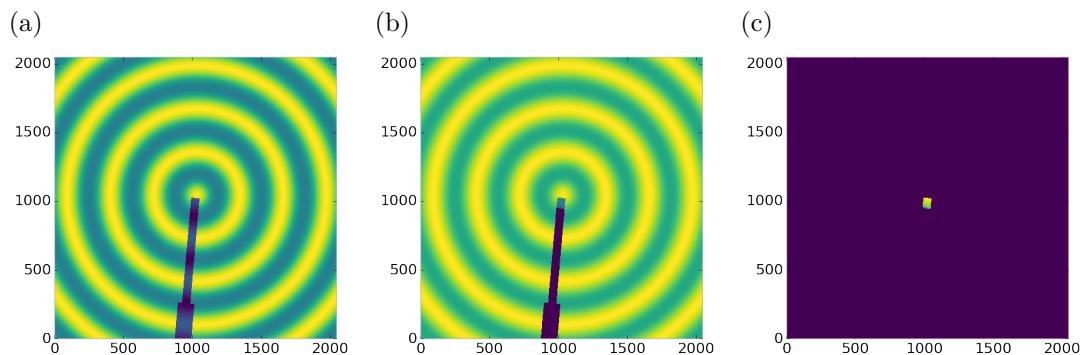


Figure 3.11: Generated beamstop holder masks which is rotated away from verticle

which contain valuable signal. This overmasking is caused by pixels being improperly associated with one another by the even bins. Figure 3.3 indicates that the masking algorithm, with the proper binning, masks the image perfectly, with no missed bad pixels or good pixels masked. This is not the case in figures 3.4 - 3.6 as we can see pixels which should have been masked but were not. Despite these missed pixels no pixels were improperly masked in any of the well binned images. These test cases are actually more difficult than experimental data, as the dynamic range of most detector causes the dead/hot pixels and single crystal/texture peaks to be orders of magnitude away from the desired signal.

The beamstop holder masks shown in figures 3.7 - 3.10, which were all run with the  $Q$  resolution binning show similar results across the transmittance range, missing only a small part of the beamstop holder near the point of incidence. Near this point the beamstop holder becomes a statistically significant part of the total number of pixels in a given ring, thus it can not be masked out using a statistical search of the rings. For most PDF and XRD studies this small area can be masked automatically by masking all the pixels whose distance from the point of incidence is smaller than a given radius  $r$ , or can be neglected outright as the area is not used in the analysis or refinement. Similar results were produced for beamstop holders which were rotated away from the vericle position, as shown in figure 3.11

Working with actual experimental data, obtained at the Advanced Photon Source beamline 11-ID-B, shows the difficulty of masking images which have low phonon counts. While the masking of experimental data taken with longer exposures, consisting of 250 .2 second shots, shown in figure 3.12 provides very sharp edges to the beamstop holder, and very little extra masking beyond the occasional dead pixel, this is not the case for the single crystal data. The single crystal data is more problematic because of its short exposure time and low flux, with 500 frame at a .1 second exposure and having shrunk the beam size. The low flux is to prevent the very strong

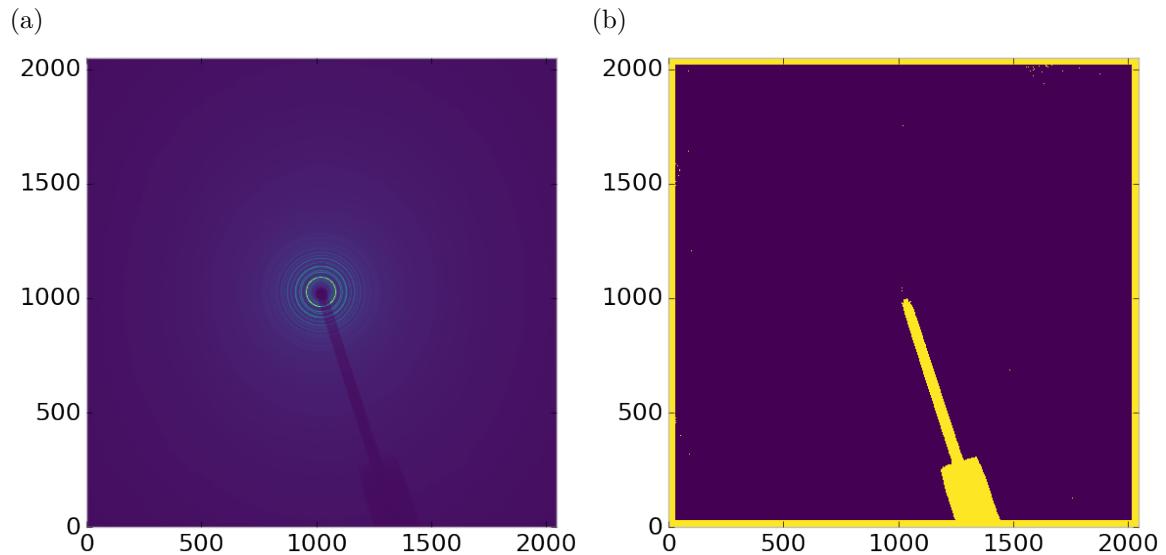


Figure 3.12: Masked experimental data. a) the raw image, b) the mask

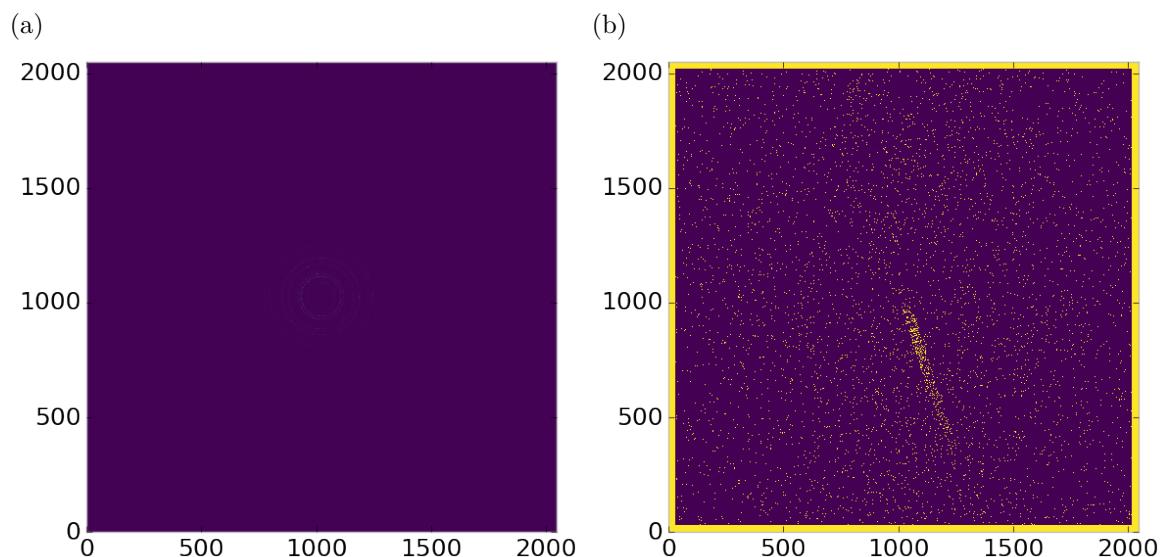


Figure 3.13: Masked experimental data with Pt single crystal signal. a) the raw image, b) the mask

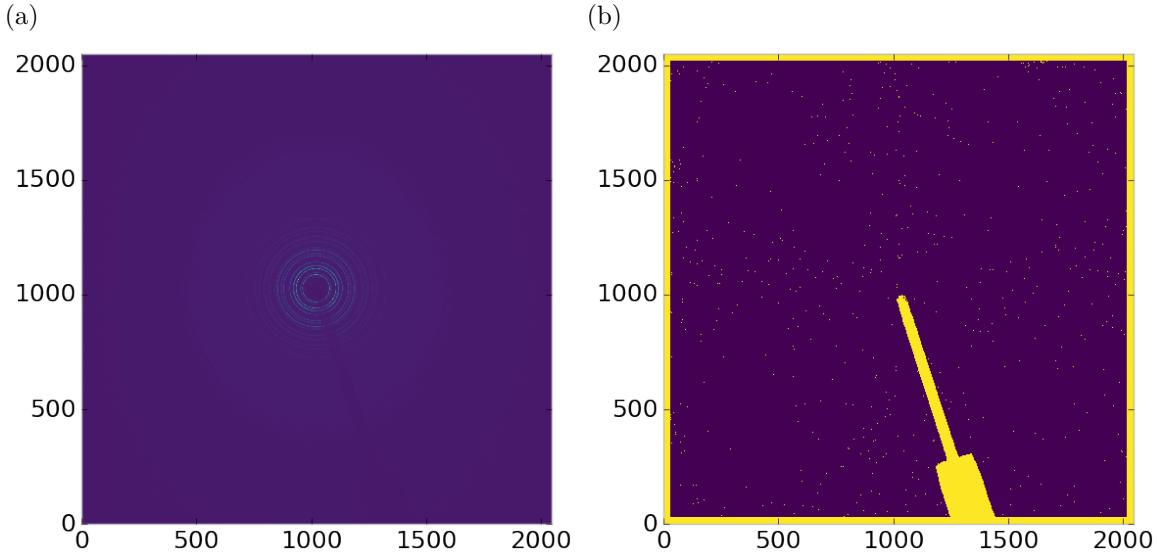


Figure 3.14: Masked experimental data with Pt single crystal signal using figure's 3.12 as a starting mask. a) the raw image, b) the mask

single crystal peaks from damaging the detector. However, this causes the image to be less statistically viable then ideal, causing problems with the mask as seen in figure 3.13. This can be alleviated to some degree by using the previously generated mask as a starting mask for the single crystal image, as shown in 3.14. While the masking algorithm still produces many diffuse masked pixels, they are far fewer, this may be due to the removal of the beamstop which could have contributed to the large standard deviation in figure 3.13.

### Conclusions

In this section the masking algorithm, which relies on both  $Q$  resolution based binning and a statistical approach to azimuthal symmetry, was developed. The focus of this algorithm was to remove many unwanted detector features associated with pixel defect, beamstop holder associated scattering attenuation, and single crystal/texture peaks. Simulated data was used to evaluate the beamstop holder and dead/hot pixel masking capacity, while experimental data was used to check for single crystal and texture based masking.  $Q$  resolution based binning was shown to be very important to

342 avoid overmasking. The ability of the mask writer to mask images is somewhat limited  
343 by the overall statistical image quality, although some deficiencies can be obtained by  
344 using previously generated masks as starting points. This masking algorithm is now  
345 in use in the data processing workflow and will be available in scikit-beam soon.

346

## CHAPTER 4

347

### BENCHMARKING

348 4.1 PDF

349 **Au55: surface relaxed**

350 **Au55: surface disordered**

351 **Au55: amorphous**

352 **Au102: triple phase**

353 **Au147**

354 **C60**

355 4.2 PDF WITH ADPs

356

## CHAPTER 5

357

### ANNEALING AND AGGREGATION OF 2NM

358

#### AU NANOPARTICLES

359 5.1 EXPERIMENTS

360 NP Synthesis

361 X-ray Total Scattering Measurements

362 5.2 DATA PROCESSING

363 5.3 DATA ANALYSIS

364 5.4 SIMULATION

365 5.5 STRUCTURAL ANALYSIS

366 5.6 CONCLUSIONS

367

## CHAPTER 6

368

### PHASE CHANGES AND ANNEALING DYNAMICS OF

369

### $\text{Pr}_2\text{NiO}_4$ AND ITS DERIVATIVES

370 6.1 EXPERIMENTS

371  **$\text{Pr}_2\text{NiO}_4$  Synthesis**

372 **X-ray Total Scattering Measurements**

373 6.2 DATA PROCESSING

374 6.3 DATA ANALYSIS

375 **Intra Sample Comparison**

376 **Inter Sample Comparison**

377 6.4 SIMULATION

378 **Small Box**

379 **Large Box**

380 6.5 STRUCTURAL ANALYSIS

381 6.6 CONCLUSIONS

382

## CHAPTER 7

383

## CONCLUSION