2025-08-06

Extracting Magnetic Exchange Interactions from Magnetic Pair Distribution Function Data via Monte Carlo Simulations

Emma Zappala
Brigham Young University

Extracting Magnetic Exchange Interactions from Magnetic Pair Distribution Function Data via Monte Carlo Simulations

Emma Zappala

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Benjamin Frandsen, Chair
Traci Neilsen
Richard Vanfleet

Department of Physics and Astronomy

Brigham Young University

ABSTRACT

Extracting Magnetic Exchange Interactions from Magnetic Pair Distribution Function Data via Monte Carlo Simulations

Emma Zappala
Department of Physics and Astronomy, BYU
Master of Science

The magnetic pair distribution function (mPDF) is a powerful tool used to analyze diffuse magnetic scattering data in real-space. Presently, mPDF is usually used to model real-space spin configurations of materials, offering important insights into the local magnetic structure, but not to directly probe the magnetic interactions giving rise to the magnetic structure in the first place. Using Monte Carlo simulations, a robust algorithm to simulate the magnetic structure of materials, we can connect the underlying magnetic exchange parameters to the spin configurations from experimental mPDF data. We prove this concept using a simple two-dimensional Ising model and then demonstrate its utility by applying it to MnTe, an important magnetic semiconductor with applications in spintronics, thermoelectrics, and more. This approach successfully captures the exchange interactions in MnTe with reasonable accuracy when compared to more standard approaches such as inelastic neutron scattering. We anticipate the extraction of magnetic interactions directly from mPDF data to be valuable in materials where conventional methods cannot be easily applied, such as geometrically frustrated magnets with disordered magnetic ground states.

Keywords: magnetic pair distribution function, mPDF, Monte Carlo simulations, magnetic exchange interaction, short range correlations, calculating exchange parameters

ACKNOWLEDGMENTS

I am so thankful for all of my teachers throughout my college career for bringing me to the place that I am. I would like to specially thank my advisor, Ben Frandsen, for encouraging and mentoring me through some of the hardest times of my life and for always believing in me and giving me a path to success for the goals I've had for my career. I am also grateful to my family and closest friends for their support throughout my education and for their willingness to lend an ear to my struggles, especially for pretending to understand whatever physics I was discussing.

Contents

Chapter 1

Introduction

1.1 Short Range Correlations

A primary focus of condensed matter physics is to understand how the microscopic structure of materials gives rise to their macroscopic properties that we observe in the real world. Scientists have dedicated their entire lives to categorizing the atomic-scale structure of materials, enough to give the study of the average crystal structure of materials a name—crystallography. However, even something as commonplace as water ice cannot be fully described by the average structure that classical crystallography would give [1], because the orientation of water molecules in the crystal structure of ice is not periodic. Instead, liquid water contains small patches of local order together where correlations are seen for a small amount of space before the correlation is lost. This type of short-range order appears in many different types of crystalline materials and, as illustrated in Figure 1.1, can be considered as small patches of spatial correlations that do not propagate through to the long-range crystal structure and thus do not contribute to the Bragg peaks in a diffraction experiment [2]. Although they are small, these regions of local structure within a material can matter just as much as the average structure for their effect on observed properties in a material [3–9].

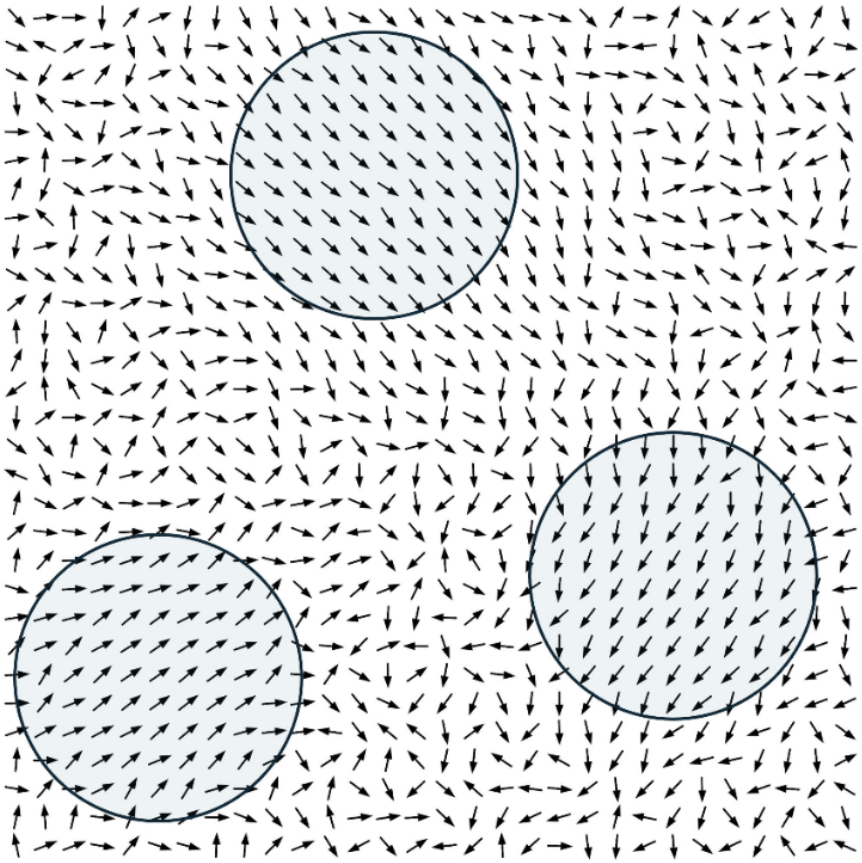

Figure 1.1 Example of short range correlations on a 2-D lattice. The circled regions show areas of local magnetic order, although other types of short-range correlations could also be represented, such as ferroelectric or ferroelastic type correlations. The areas outside the circle fall away into random order as short-range order is lost. While these areas all fall into roughly ferromagnetic configurations, they are oriented in different directions as these areas of order do not affect each other. Figure from Frandsen and Fischer [2].

A myriad of materials have interesting properties due to the short-range order within it. For example: some corrosion-resistant alloys [10], the thermodynamics of high entropy alloys [11], efficiency and performance in Li-ion battery materials [12], and the size of band gap in some semiconductors [13] are all different macroscopic behaviors of materials that arise because of short-range correlations within the material's crystal structure.

The same way that short range physical order helps us understand physical properties of materials, short-range magnetic order, or orientational correlations among spins that are well-defined over a local region on the atomic lattice and average away to zero on longer length scales, helps us understand magnetic properties of materials. Short-range magnetic order is exhibited in a number of different magnetic materials. Materials like geometrically frustrated magnets [14, 15], spin glasses [16], and spin liquids [17] are all materials dominated by short range magnetic correlations and lack long range magnetic order. Many materials with long-range magnetic order also show short-range order above the magnetic transition temperature, which can also impact important material properties such as electrical or thermal conductivity [2]. Understanding short-range magnetic correlations in materials is imperative to expanding our fundamental understanding of magnetism and how it affects technologically important properties.

1.2 Atomic and Magnetic Pair Distribution Function Analysis

We can probe the short-range features of a material, including the structure and the role played by charge, orbital, and spin degrees of freedom, through techniques like X-ray and neutron scattering. In particular, "total scattering" techniques [5, 18–20] utilize both Bragg scattering and diffuse scattering intensity to probe the long-range and short-range structural correlations in materials, respectively. In the case of neutrons, the scattered intensity originates both from the nuclei in the material and any magnetic moments present due to unpaired electrons. As a result, both the atomic and magnetic structure of a material can be investigated from neutron scattering data.

The pair distribution function (PDF) is a popular and powerful total scattering technique to observe the atomic structure of materials. The PDF involves Fourier transforming scattered X-ray or neutron intensity, transforming scattering data from momentum space into real space to reveal the local pairwise correlations directly [21]. Examining a PDF for a given material gives real-space

information about both long range and short range ordered atomic structure [22]. A larger PDF magnitude at a particular interatomic distance r, corresponds to a higher the probability that there is another atom that distance from some arbitrary reference atom [22]. By examining the PDF pattern across varying real-space ranges, it is possible to separate long-range structure and short-range correlations, giving a more complete view of the structure of a material than would be available from conventional Bragg diffraction techniques.

Similarly, the magnetic pair distribution function (mPDF) takes advantage of magnetic neutron scattering in materials in order to understand long- and short-range magnetic order in a material. Magnetic PDF was formalized by Frandsen *et al.* in 2014 [23]. Orientational correlations between pairs of magnetic moments result in magnetic scattering of neutrons, which can either take the form of sharp Bragg peaks in the case of long-range magnetic order or diffuse scattering in the case of short-range correlations. Magnetic scattering is most easily seen in neutron scattering data, as x-rays are only weakly sensitive to magnetism compared to charge scattering. The magnetic contribution to the neutron scattering pattern is Fourier transformed to produce the mPDF, yielding information about real-space correlations between magnetic moments [23]. Typically, mPDF patterns are analyzed qualitatively to infer the local magnetic structure and/or fit quantitatively with a model of the local spin configuration in the materials. As is be demonstrated in this thesis, mPDF patterns can also be analyzed to yield information about the underlying magnetic interactions that produce the observed local magnetic structure, but this type of mPDF analysis has been underutilized to this point.

1.3 Magnetic Exchange Interactions and the Hamiltonian

In most quantum materials, magnetic order is governed by magnetic exchange interactions, which originate from a combination of the Pauli exclusion principal and the Coulomb repulsion between

electrons. Exchange interactions are the driving force not only for magnetic transitions to long-range order at the Curie or Néel point, but also for the existence of short-range magnetic correlations in materials [24].

An electron's wavefunction depends on its spatial coordinates and its spin state. When two or more electrons interact, their wavefunctions combine to form a product state. Because electrons are fermions subject to the Pauli exclusion principle, the overall wavefunction must be antisymmetric; to achieve this, a symmetric spatial wavefunction requires an antisymmetric spin wavefunction and vice versa. To illustrate this in the simplest possible case, consider just two electrons. The two states allowed for a set of two electrons are either a singlet state, consisting of a symmetric product of spatial wavefunctions and an antisymmetric product of spin wavefunctions, or a triplet state, consisting of an antisymmetric spatial wavefunction and a symmetric spin wavefunction. The singlet state can be considered a type of antiferromagnetic state, since the spins are anti-aligned, while the spins are aligned in the case of the triplet state, similar to ferromagnetism.

In more detail, two spin-$\frac{1}{2}$ particles have the interaction

$$\hat{H} = A\hat{\mathbf{S}}^{\mathbf{a}} \cdot \hat{\mathbf{S}}^{\mathbf{b}} \tag{1.1}$$

where $\hat{\mathbf{S}}^{\mathbf{a}}$ and $\hat{\mathbf{S}}^{\mathbf{b}}$ are the spin operators for the two particles [25]. Combining the spins of these two particles gives a total spin s of either 1 or 0. As a result, the Hamiltonian has two different energy levels, where

$$E = \begin{cases} \frac{A}{4} & \text{if } s = 1 \\ -\frac{A}{4} & \text{if } s = 0. \end{cases}$$

the $s = 0$ case is a singlet, and the $s = 1$ case is a triplet. The difference between singlet and triplet states in a material makes up the effective Hamiltonian for this example

$$\hat{H} = 1/4(E_S + 3E_T) - (E_S + E_T)\hat{\mathbf{S}}_{\mathbf{1}} \cdot \hat{\mathbf{S}}_{\mathbf{2}}, \tag{1.2}$$

where E_S is the energy of the singlet state, and E_T the energy of the triplet state [25]. This Hamiltonian has a spin-dependent term and a spin-independent term. The spin-independent term is just a constant, and does not give much useful information about the system, but the spin-dependent term determines whether the material prefers singlet or triplet states, or antiferromagnetic or ferromagnetic behavior. The constant spin-independent term can be omitted, leading to the Heisenberg model for magnetic spins, which takes just the spin-dependent term of the Hamiltonian, simplifying the Hamiltonian to

$$\hat{H} = \sum_{ij} J_{ij} \hat{\mathbf{S}}_\mathbf{i} \cdot \hat{\mathbf{S}}_\mathbf{j}, \tag{1.3}$$

or the sum of all the individual interactions between all of the electrons in a material. If $J < 0$, the system prefers the singlet state, or antiferromagnetic behavior, and if $J > 0$ the system prefers the triplet state, or ferromagnetic behavior. The different values for J are usually grouped in a material into the corresponding near-neighbor coordination shells. As the distance between atoms increases, the J value also decreases, so it is unnecessary to use all of the J values to describe the magnetic order in a material; just the exchange interactions of the first few neighbors are needed. It is rare to see more than the four nearest neighbors used to make a model of the magnetic order for a material.

The simplest form of magnetic interaction is direct exchange, where neighboring magnetic atoms orbitals directly interact with each other. However, most real-life materials are not as simple as just a direct exchange between electron orbitals in atoms. Often other electrons from other atoms that interfere, referred to as indirect exchange. Materials often have a mix of magnetic and non-magnetic atoms, separating the electron orbitals of the magnetic atoms to a greater distance, with an intermediary non magnetic orbital in between. The main material addressed in this paper, MnTe, exhibits such behavior. In Figure 1.2, showing the atomic and magnetic structure of MnTe, There are alternating layers of Mn, the magnetic atom in this material, and Te, the non-magnetic atom in this material. This type of indirect exchange is called superexchange [25]. A set of rules called the Goodenough-Kanamori rules help determine the whether a material experiencing superexchange

will be ferromagnetic or antiferromagnetic. They are based on two main parts of superexchange, a potential term based in electron repulsion (a ferromagnetic contributor) and the second a kinetic term that dominates (an antiferromagnetic contributor). A non-magnetic orbital between the two magnetic orbitals causes the outer electrons of the magnetic material to delocalize over a wider range, lowering the kinetic energy [25]. As a result, the angle of the bond between the magnetic and non-magnetic atoms is the most important factor in the Goodenough-Kanamori rules to determine the magnetic order of a system.

The traditional method of experimentally solving for the exchange interaction in real materials involves measuring the inelastic neutron scattering of a material below its ordering temperature, then building an interaction model to fit the observed spin waves [26, 27]. Observing the spin waves in materials is highly effective when the material possesses conventional long-range-ordered magnetic ground states, but this method cannot be applied to materials do not experience any long-range order. Materials like spin liquids and spin gasses cannot have their exchange interactions quantified in this way [28–30]. In addition, if the exchange interactions are different below and above a long-range magnetic phase transition (due to, for example, a simultaneous structural phase transition), this method is unable to measure the exchange interactions above the transition. Recently, utilizing magnetic diffuse scattering has emerged as a viable option for determining magnetic exchange interactions in cases where long-range magnetic order is not present. The software Spinteract matches diffuse scattering to mean-field-theory-based models of spin Hamiltonians [31]. Another approach is to use mean field theory to extract exchange parameters from diffuse magnetic scattering data [32]. Our goal is to demonstrate the feasibility of using mPDF data to determine magnetic exchange interactions via Monte Carlo simulations, a well-established method to simulate magnetic configurations [33]. The potential advantages in this approach are that (1) it is another option for determining magnetic interactions when traditional spin-wave analysis is not possible, including in correlated paramagnetic states above long-range transitions; (2) it can be done simultaneously with

traditional mPDF analysis and/or atomic PDF analysis when the local atomic structure is also of interest; and (3) exchange interactions are most easily described in real space, making the real-space mPDF a natural quantity from which to determine the interactions.

1.4 The 2-D Ising Model

The two-dimensional (2-D) Ising model provides a useful testing ground for our investigation of Monte Carlo simulations and mPDF analysis. The Ising model is a classic and simple system comprised of a lattice of interacting magnetic spins; for our 2-D Ising model, we choose two-dimensional square lattice. Ising spins take one of two values: spin up (1) or spin down (-1). No long-range transition occurs for the 1D Ising model (a 1D grid of spins), but the 2-D Ising model does undergo a transition. This system was solved in its entirety in 1944 by Lars Onsager [34]. It has since been well documented both theoretically and numerically [35–38]. The derivation is omitted here, but the salient results are mentioned. First, for a 2-D lattice of spins, the critical temperature is

$$T_c = \frac{2J}{k_B \ln(1+\sqrt{2})}. \tag{1.4}$$

When J = 1 K/k$_B$, this simplifies to $T_C = 2.269$ K [37]. The two-point correlation function, which describes the statistical correlation between two points on a lattice is defined as

$$\Gamma(r_i - r_j) = \langle S_i S_j \rangle - \langle S_i \rangle \langle S_j \rangle, \tag{1.5}$$

where r_i and r_j are two positions in the lattice and $\langle S_{i/j} \rangle$ refers to the spin vector for the i/j-th position. These particular results assume an infinite lattice, where the translational invariance correlation only depends on the distance r instead of the absolute position of the sites. At high temperatures, the statistical correlations between any r are lost, but as temperature decreases and the spins start to align $\Gamma(r_i - r_j)$ increases.

In the regime above the magnetic transition ($T > T_C$), $\langle S_i \rangle$ is zero. As a result the statistical correlation between two points simplifies to

$$\Gamma(r_i - r_j) = \langle S_i S_j \rangle, \tag{1.6}$$

which is approximately equal to $e^{-|r_i - r_j|/\xi(T)}$, where ξ is the correlation length for a given temperature. The exponential dependence for ξ can be solved to gives the function for the correlation length ξ above T_C. The correlation length is the parameter usually used to determine the strength of short-range correlations within a material. The correlation length approaches zero in the limit of very high temperature, or $T \gg T_C$ (when short range correlations die out); upon cooling toward the critical temperature, the correlation length increases exponentially, until it reaches a functionally infinite amount when $T < T_C$.

While the 2-D Ising model is exceptionally simple and many materials are much more complex, some materials that can be modeled as 2-D Ising systems, especially if there is anisotropy in the material, making the spins favor an up and down orientation [39, 40]. For example, $TmMgGaO_4$ [41] (triangular lattice) and atomically thin $FePS_3$ [42] (honeycomb lattice) are dominated by antiferromagnetic Ising behavior, and Cr_2O_3 (kagome-honeycomb lattice) displays ferromagnetic Ising behavior [43]. Despite its simplicity, the 2-D Ising model remains a useful tool for physicists, and thus is a notable model to assess the validity of our simulations.

1.5 MnTe

To investigate the efficacy of our new technique, we applied it to MnTe, an important binary semiconductor that has garnered substantial research interest in the condensed matter community. Its incredibly interesting magnetic [44–46], electronic [47], and optical [48] properties have marked it as an attractive candidate for potential technological applications. Its thermoelectric [49] properties are also noteworthy, particularly in the context of short-range magnetic correlations. MnTe experiences

paramagnon drag, whereby short-range magnetic correlations in the paramagnetic state enhance the Seebeck coefficient and boost the thermoelectric figure of merit [50, 51], making it possible to use MnTe to recover energy from heat waste [52]. MnTe has most recently gained notoriety as a potential altermagnet [53], a new class of material that breaks time-reversal symmetry, similar to a ferromagnet, but does not generate a net magnetic moment, similar to an antiferromagnet. Altermagnets have the potential to enable groundbreaking advancements in spintronics.

MnTe is a type-A antiferromagnetic material with a Néel temperature of around 307 K [44, 54]. Its $S = 5/2$ Mn^{2+} spins have parallel alignment within the *ab* planes along the $\langle 1\bar{1}0 \rangle$ direction, with antiferromagnetic coupling along the *c* axis, as shown in Figure 1.2. The magnetic structure of MnTe can be described with a Heisenberg spin Hamiltonian

$$\hat{H} = -\sum_{ij} J_{ij} \mathbf{S}_i \cdot \mathbf{S}_j - \sum_i K |S_i^z|^2 \tag{1.7}$$

where J_{ij} is the exchange constant between the *i* and *j* spins, $\mathbf{S}_i$ is the spin vector of the *i*th atom, and *K* is the single-ion magnetic anisotropy. The single-ion mangetic anisotropy describes the tendency for some lattices to align their mangetic spins in a preferred plane (in MnTe this is the a/b plane, as shown in Figure 1.2). The crystalline field inside a material interacts with the electron orbitals and can quench the moments, causing spin orbit coupling between the orbitals—aligning them with the lattice. Specifically, the values for the first four nearest exchange interactions, $J_1 = -4.125$ meV, $J_2 = -0.025$ meV, $J_3 = -0.55$ meV, and $J_4 = -0.2175$ meV [55] (also shown in Figure 1.2), are sufficient to describe the magnetic ordering of this material. J_1 is the strongest of these and primarily dictates the antiferromagnetic behavior of the material. Additionally, J_2 and J_4 are also negative, which promotes anti-alignment in the nearest neighbors that are aligned in the wider antiferromagnetic lattice. Their relative weakness in comparison to J_1 leaves their anti-alignment tendencies unrealized.

Due to its highly studied nature, MnTe has copious amounts of mPDF characterization [2, 52, 55]. Figure 1.3 shows alternating patches of anti-alignment and alignment consistent with

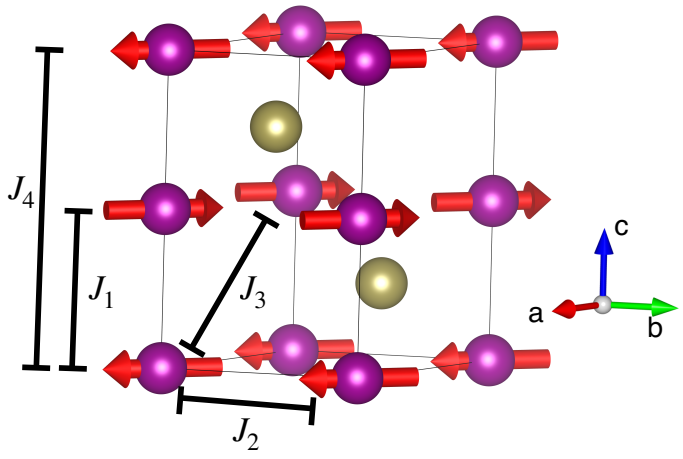

Figure 1.2 Structure of MnTe. Mn atoms are shown as purple, and Te atoms are shown as gold. The direction of each Mn spin is shown with a red arrow. Each *ab* plane aligns ferromagnetically with antiferromagnetic alignment across planes on the c axis. Each applicable near-neighbor interaction is noted with J_1, J_2, J_3 and J_4.

antiferromagnetic behavior, but they die out fairly quickly. This particular set of data is taken at 307 K, well above the transition temperature, and the dying out of the magnetic correlations shows the short-range correlations seen in MnTe above the magnetic transition. Short range correlations can also be seen in the 2-D mPDF in Figure 1.4 A, as magnetic correlations persist even at 320 K, even further above the magnetic transition. MnTe's well characterized and persisting short-range correlations make it an excellent use-case candidate for Monte Carlo simulations to extract magnetic exchange interactions.

The experimentally measured ordered magnetic moment for MnTe are shown in Figure 1.4 B. The blue triangles are from a shorter fitting range, representing the ordered moment for nearest-neighbor spin pairs, while the orange circles are from a longer fitting range representing the true long-range order of MnTe. The persistence of the short-range magnetic order parameter above the Néel temperature shows that significant antiferromagnetic correlations between nearest neighbor moments remain up to high temperatures. The correlation length versus temperature for MnTe in the

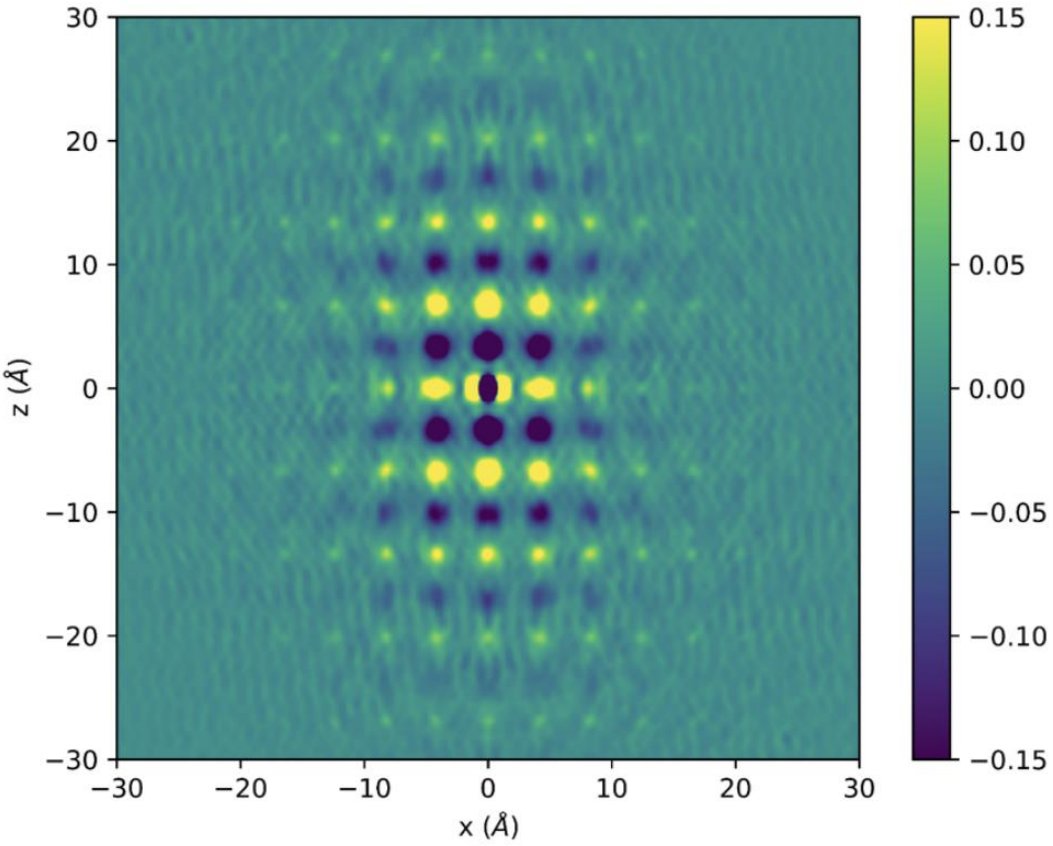

Figure 1.3 3D-mPDF pattern, featuring antiferromagnetic short-range correlations with an anisotropic correlation length: yellow indicates ferromagnetic interactions and the deep blue antiferromagnetic. This example graph is in the x-z plane of MnTe, with $T \sim 340$ K. The units shown in the color bar are arbitrary. Figure taken from Baral [56]

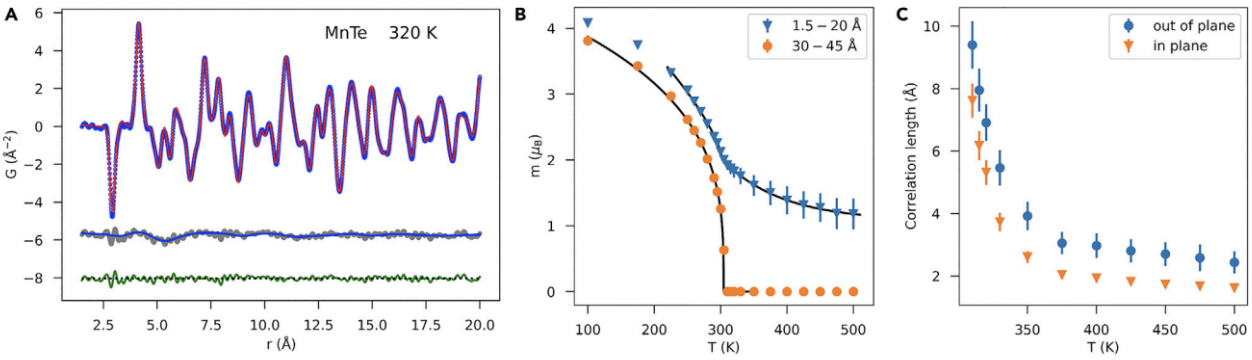

Figure 1.4 (A) combined atomic and magnetic PDF fit for MnTe where $T = 320$ K (above the Néel temperature). The gray curve highlights the short range correlations at this temperature. The green gives the residual from subtracting the fit from the data. (B) The ordered magnetic moment of MnTe as determined from the mPDF fits at various temperatures with different fitting ranges. The blue triangles are from fits made with 1.5–20 Å while the orange circles are from fits made with 30–45 Å fitting range. (C) The fitted correlation length from the 1.5–20 Å fits. Figure taken from Baral [52].

paramagnetic phase is shown in Figure 1.4 C. The out-of-plane correlation length remains longer than the in-plane correlation length, which can be understood as a consequence of the stronger exchange interaction along the *c* axis than within the *ab* plane. Both increase as the temperature decreases, showing a rapid divergence toward long-range order in the immediate vicinity of the transition temperature at 307 K.

Chapter 2

Methods

2.1 The Theory of Magnetic PDF

Magnetic PDF is a powerful tool that allows insight into the local-scale magnetism of a material. It involves Fourier transforming the magnetic component of neutron scattering data. The equation for the mPDF for a material with a single type of magnetic atom with spin-only magnetic moments is

$$\begin{aligned} G_{\mathrm{mag}}(r) &= \frac{2}{\pi}\int_{Q_{\mathrm{min}}}^{\infty} Q\left(\frac{\left(\frac{d\sigma}{d\Omega}\right)_{\mathrm{mag}}}{\frac{2}{3}N_s S(S+1)(\gamma r_0)^2[f(Q)]^2} - 1\right)\sin(Qr)\,dQ \\ &= \frac{1}{N_s 2S(S+1)}\frac{3}{r}\sum_{i\neq j}\left(A_{ij}\delta(r-r_{ij}) + B_{ij}\frac{r}{r_{ij}^3}\Theta(r_{ij}-r) - 4\pi r\rho_0\frac{2}{3}m^2\right). \end{aligned} \tag{2.1}$$

The first of these equations is the experimental definition of the mPDF [57]. The integration covers the different magnetic scattering vectors Q, from the minimum measured scattering vector Q_{min} (Q_{min} must be large enough so it is beyond the small-angle scattering regime), to infinity. In practice, the Fourier transform is computed only up to a finite maximum Q value, but it should be as large as possible while keeping the signal-to-noise ratio greater than unity. The integrand is made up of the magnetic scattering cross section $\frac{d\sigma}{d\Omega}(q)$, the real-space distance r, the classical electron radius r_0 where $r_0 = \frac{\mu_0}{4\pi}\frac{e^2}{m_e}$, the quantum number S (with units of $\hbar$), the neutron magnetic moment in

units of nuclear magnetons $\gamma = 1.913$, the magnetic form factor $f(Q)$, and the number of spins in the system N_s. The second of these equations is the theoretical definition of the mPDF defined for a specific magnetic model, i.e., the configuration of positions and orientations of spins. As a whole the second equation in Equation 2.1 takes the sum over individual spins $\mathbf{S}_i$ and $\mathbf{S}_j$ separated over some distance r_{ij}, where $A_{ij} = \langle S_i^y S_j^y \rangle$, $B_{ij} = 2\langle S_i^x S_j^x \rangle - \langle S_i^y S_j^y \rangle$ (given in the coordinate system $\hat{\mathbf{x}} = \frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|}$ and $\hat{\mathbf{y}} = \frac{\mathbf{S}_i - \hat{\mathbf{x}}(\mathbf{S}_i \cdot \hat{\mathbf{x}})}{|\mathbf{S}_i - \hat{\mathbf{x}}(\mathbf{S}_i \cdot \hat{\mathbf{x}})|}$), Θ is the Heaviside step function, p_0 is the number of spins per unit volume, and m is the average net magnetic moment per spin in μ_B [57], which is zero for antiferromagnets. Examining Equation 2.1 gives intuition behind the meaning of an mPDF plot. Peaks in the plot arise at distances r where there are spin-spin correlations; where there are different neighbors are on the atomic lattice [2]. The sign of the peak determines whether or not that correlation is ferromagnetic or antiferromagnetic. As a result, the mPDF is a simple way to visualize and determine the spin configuration inside a material.

There are two main ways to generate experimental mPDF data. One is just directly using Equation 2.1, which entails dividing by the squared magnetic form factor before taking the Fourier transform. This effectively deconvolves the mPDF pattern from the size effects of the spin-density distribution in real space, resulting in a sharp, high-resolution mPDF pattern in real space that can be advantageous when fitting magnetic models. We call this the "normalized" or "deconvolved" mPDF pattern. However, this is often prone to errors and artifacts, because it requires dividing potentially noisy scattering intensities at high Q by the squared magnetic form factor, which is very small at high Q and thus may artificially amplify the noise. If the data quality is not high enough to meet the stringent demands of the normalized mPDF, then one can also skip the normalization by the squared magnetic form factor and simply Fourier transform the product of Q and the magnetic scattering intensity, which results in the "unnormalized" or "non-deconvolved" mPDF. This is much easier to obtain experimentally and suffers from much less noise, but it also has greatly reduced real-space resolution. The unnormalized mPDF can be thought of as the properly normalized mPDF

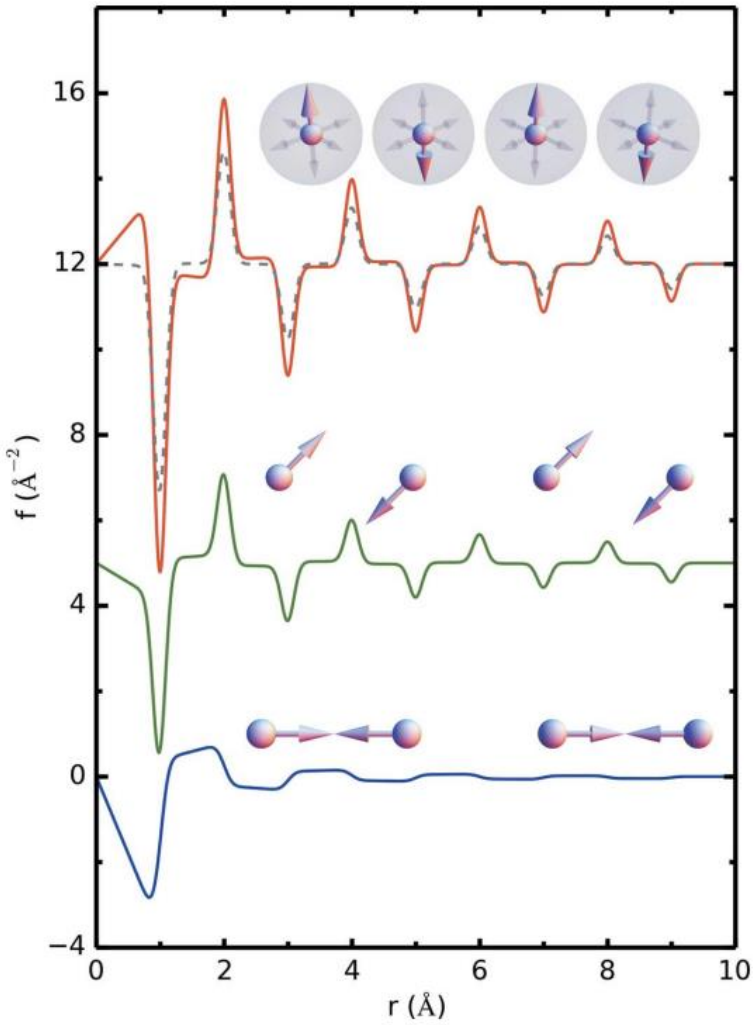

Figure 2.1 Theoretical normalized mPDFs from various antiferromagnetically coupled spins, each in a different orientation shown next to the graphed mPDF. The broken gray line is the orientationally averaged case. Figure taken from Frandsen [23].

broadened out in real space by approximately $\sqrt{2}$ times the real-space extent of the spin-density distribution. In this work, all experimental data is unnormalized mPDF data, but the simulated data takes both forms. Each form will be noted.

An illustrative example of simulated no4malized mPDF data for one-dimensional spin chains is given in Figure 2.1.The x-axis represents some arbitrary distance r from some reference atom.

Peaks appear at distances separating two spins from each other; since the different chains of spins have the same inter-spin distance, features occur at the same r values in the graphs. Each has alternating negative and positive peaks, showing that the atoms in the chain alternate from aligning and anti-aligning from the reference atom, indicative of antiferromagnetic order. The mPDF is highly sensitive to the orientations of the spins, as seen in the differences between the mPDFs for the same spaced chains with different spin orientations. This sensitivity to orientation helps make mPDF a very useful tool for determining spin orientations in materials.

2.2 Monte Carlo Simulations

Monte Carlo simulations were first used to simulate phase transitions between liquids and solids, but they have also been used extensively to simulate magnetic transitions [33]. This technique is based in probability [58], making it uniquely suited to simulate magnetic transitions. We used the Metropolis algorithm, a subset of Monte Carlo simulations, to simulate the magnetic structure of the given material. We introduce this algorithm using the 2-D Ising model, to which the Metropolis Monte Carlo method has often been applied to find the transition temperature of the 2-D Ising model. Take a simple square lattice made up of particles with spins s_i, either up or down, and configure the spins randomly on the lattice. In the Metropolis algorithm, a spin s_i is selected at random and its spin is flipped, constituting a "trial move". The change in energy associated with flipping the spin is

$$\Delta E = -\sum_{ij} J_{ij} s_i s_j, \tag{2.2}$$

where J_{ij} indicates the exchange interaction between spins s_i and s_j. These sums are limited to the first four nearest neighbor interactions, as any magnetic interactions beyond that are almost always negligible. If $\Delta E \leq 0$, the spin-flip reduces the energy, and is therefore accepted into the system. However, if $\Delta E > 0$, then the Metropolis algorithm still sometimes accepts the "bad move" even though it increases the energy of the system. The change is accepted if and only if $e^{(-\Delta E/k_B T)} > x$,

where x is some random number generated between zero and one. A random x is generated for each energy-raising move. Using the exponential and a random x introduces an acceptance of "bad moves" into the system; moves that raise the total energy only slightly have a higher chance of being accepted than more costly moves. In addition, higher temperatures result in a higher probability of accepting a bad move than lower temperatures. This introduced randomness into the magnetic structure helps simulate the effect of thermal fluctuations, and the general randomness that comes with real life materials. It also reduces the chance of being caught in a local minimum while exploring the energy parameter space. Each trial spin-flip is referred to as a "Monte Carlo step".

This same method can be applied to any material, not just the 2-D Ising model. All that is needed is the atomic structure and the set of near-neighbor exchange interactions J for the given material. Structural information for a material can be found in a crystallographic information file (.cif), for which large databases can be accessed that contain hundreds of thousands of .cif files for nearly every material that has been reported in the scientific literature. The process for the Metropolis algorithm is the exact same as for the 2-D Ising model, but the positions of atoms, and value of spin is different. The Ising spin takes only 'spin up' and 'spin down' values, and a lot of materials do not behave in this way. People also describe spins with the X-Y model, where spins are restricted to a certain plane and are free to move around in any direction within that plane. The most generalized model, the Heisenburg model, allows spins to point in any direction. The regime of spins used depends on the anisotropy (or lack of) in a material. Some materials, like those described in Section 1.4, have anisotropy that heavily favors Ising spins, while some, like MnTe, have planar anisotropy. Those that are isotropic favor Heisenburg spins. Since it is the most generalized, the Heisenburg model also serves as a 'catch all' if the material's anisotropy is unknown. We used the Heisenburg model in our generalized simulations (even though MnTe experiences planar anisotropy) for this purpose.

The main downside to using such a generalized model, however, is the long computation time needed to compensate for the generalization. Monte Carlo simulations of real materials often take tens of thousands of steps or more, many more than would normally be required for the Ising model. When working with Heisenberg spins, a single step chooses a new spin in a random direction to compare to the existing spin in the lattice. Because there are so many different orientations for the spin to take, instead of just up or down, or confined to a plane like the x-y model, many more Monte Carlo steps are needed to find an energy equilibrium. If the anisotropy of a material is known, it is generally recommended to use the appropriate model to run Monte Carlo simulations to limit computation time.

2.3 Using Monte Carlo to Simulate the mPDF

The goal of this project is to use Monte Carlo methods to simulate mPDF data and infer useful information about the underlying magnetic exchange interactions, either by inspection of the simulation results themselves or through comparison to experimental data. I now describe in detail our implementation of Monte Carlo to simulate mPDF patterns using the Python programming language. Consider a lattice of random spins—-as an example, the square lattice of the 2-D Ising model shown in Figure 2.2a. Once the atomic structure is built all the relevant nearest neighbors for each individual spin are found and stored in a Python dictionary. Then run the Metropolis algorithm on that lattice as outlined in Section 2.1, starting at high temperature (many times larger than the energy scale of the interactions). The simulation runs until equilibrium has been reached for that temperature (i.e., fluctuations of the energy are stable within k_BT). The more complex the system, the more Monte Carlo steps are needed to find that equilibrium. Once an equilibrium is established, the final spin configuration for the starting temperature is plugged into the Python package diffpy.mpdf [**?**]to generate a MagStructure object, which records the magnetic spin structure

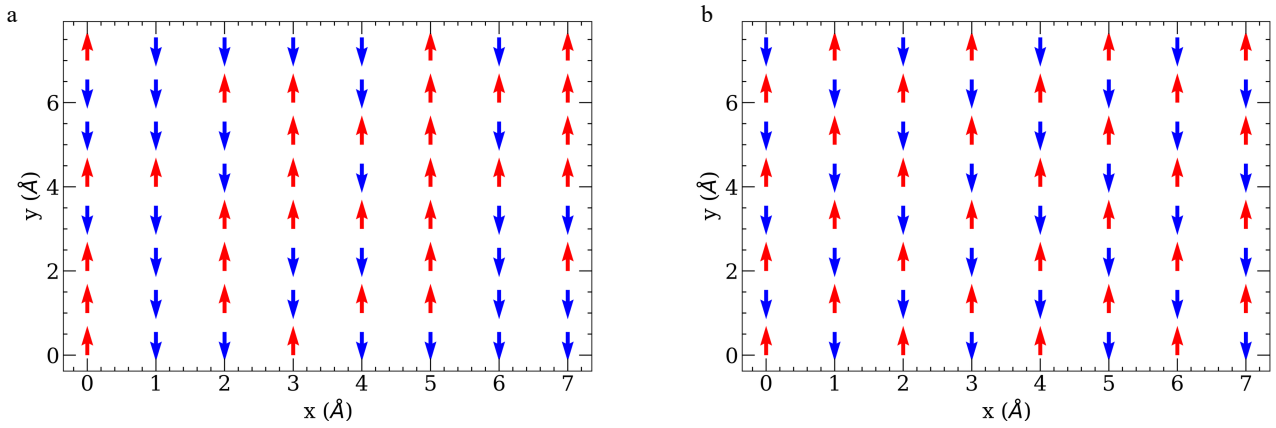

Figure 2.2 (a) Random distribution of spin up and spin down spins in an 8x8 grid. (b) The same Ising grid after a Monte Carlo simulation where J = -1 at T = 0.5 K, leading to a completely ordered antiferromagnetic distribution of spins.

of a material. This MagStructure can then be used with the diffpy.mpdf MPDFcalculator class to calculate a theoretical mPDF using Equation 2.1. All this equation needs to calculate the mPDF is the spin configuration of the material. The diffpy.mpdf package calculates the mPDF for a given MagStructure, which is then saved in a Python dictionary.

The temperature is then lowered slightly, and the algorithm is run again, except that the initial state is no longer completely random, but is the final state resulting from the previous temperature. The temperature is gradually lowered in this manner, simulating the material itself cooling in some kind of annealing process, until the initially random distribution at high temperature has transitioned to an ordered state, as shown in Figure 2.2b. This iterative approach creates a more accurate mPDF for lower temperatures without the computational expense of running Monte Carlo steps for an unnecessarily long time and avoids non-physical artifacts in the mPDF that arise when it is run only at a low temperature, where the lack of thermal energy often traps the simulation in a local minimum. After equilibrium is reached at each temperature, the mPDF is again calculated and saved, such that mPDF patterns for each temperature step are available for subsequent analysis.

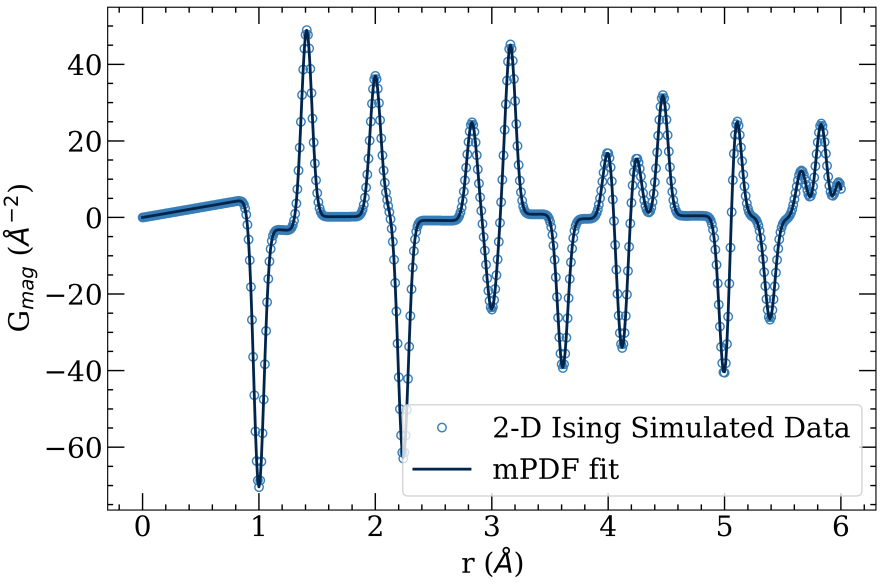

Figure 2.3 Simulated mPDF for an 8x8 2-D Ising grid with J= -1 K and T= 1 K.

An example of the final simulated mPDF for the 2-D Ising model at low temperature (after cooling from high temperature) is shown in Figure 2.3. At 1 Å, the first nearest neighbor distance for the Ising grid, there is a large negative peak, corresponding to antiferromagnetic alignment between any given spin and its nearest neighbors at a distance of 1 Å. The peak position is congruent with the spin configuration shown in Figure 2.2b, the configuration used to generate this mPDF. Then the next peak shows ferromagnetic alignment, or a large positive peak, at the next nearest neighbor distance around 1.4 Å, corresponding to the diagonal neighbor on the Ising square grid. This peak further confirms the antiferromagnetic behavior of this 2-D Ising grid. If the nearest-neighbor interactions on the grid are antiferromagnetic, the diagonal spins should be parallel. The identical agreement between the mPDFs calculated from the Monte Carlo simulation and the ideal spin arrangement shown in Figure 2.2b indicates that the simulation found the correct ground state of the 2-D antiferromagnetic Ising model.

To make our Monte Carlo simulations have a managable size with regard to computation time, we implemented a super-cell of several unit cells (for the Ising model a unit cell would just be one

spin in the corner of the cell) and used periodic boundary conditions to simulate an infinite lattice. We accomplished periodic boundary conditions by creating copies of the supercell such that the original supercell occupies the center of a 3×3 tiling of supercells (or the center of a $3 \times 3 \times 3$ tiling for the generalized 3-D version). Each change made in the original supercell due to Monte Carlo simulations is then echoed into the surrounding copies. As a result, the original supercell size sets the maximum distance over which a meaningful mPDF pattern can calculated. The mPDF cannot be calculated beyond this maximum size of the supercell, as artifacts from repeating atoms appear when calculating correlations of repeating spins. Throughout this work, we calculated our mPDFs out to a maximum value of 15 Å, which is sufficient to view the local magnetic interactions and correlations of interest for this project.

However, calculating the mPDF after just one Monte Carlo simulation is not sufficient to describe a material's true mPDF. The element of randomness inherent in Monte Carlo simulations produces non-unique solutions—-multiple different spin configurations can lead to very similar energies of the system. Despite the energies of these systems being essentially indistinguishable, the different spin configurations would result in slightly different mPDFs. To get an accurate representation of the magnetic structure for a given material, we performed 1000 individual Monte Carlo simulations and averaged the resulting mPDFs together, using a weighting scheme identical to the Metropolis algorithm's weight for 'bad moves': $e^{(E_{\text{tot}}/k_B T)}$, where E_{tot} is the total energy for the mPDF being weighted. As a result, mPDFs from lower-energy configurations get more weight when calculating the average, consistent with lower-energy systems being more likely to appear in real life.

Chapter 3

Results: mPDF Modeling of Monte Carlo Simulations

3.1 Calculating the mPDF with the 2-D Ising Grid

I used the 2-D Ising model as a test case for Monte Carlo simulations of mPDF data. To do so, I initialized a 20×20 square lattice of Ising spins, each 1 Å from its nearest neighbor. For the 2-D Ising model with only nearest-neighbor interactions, the theoretical magnetic ordering transition occurs at $2.269J/k_B$ for both ferromagnetic and antiferromagnetic interactions [59]. Using $J = \pm 1$ K leads to a simple Hamiltonian $H = \pm \sum_{ij} s_i s_j$, where the sum goes over all pairs of nearest neighbors. I modeled the system with both ferromagnetic and antiferromagnetic interactions and found no meaningful difference between the two, so I show just the antiferromagnetic results, since most of the magnetic materials studied by mPDF are antiferromagnetic. I worked through the process outlined in Section 2.3, starting at a temperature of 250 K, and working down in a simulated annealing process to each temperature point of interest.

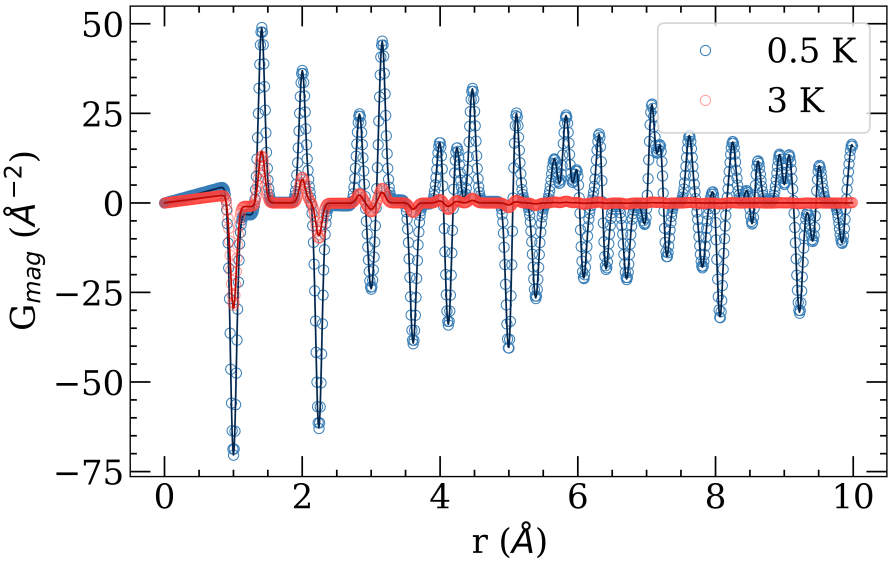

Figure 3.1 Example mPDF's and their respective fits for $T = 3$ and $0.5K$, for a 20x20 2-D Ising grid, where $J = -1$. From some arbitrary distance r, a positive G_{mag} corresponds to a ferromagnetic alignment between atoms at that distance, and negative a antiferromagnetic alignment at that distance. Antiferromagnetic behavior is clearly seen here, as well as the transition between a state with short-range correlations and an ordered state.

Example simulated mPDF data is shown for a high temperature and a low temperature in Figure 3.1. This figure highlights the difference between the mPDF at temperatures below and above the transition. Above the transition, the mPDF correlations between spins are damped to zero over some finite distance, as shown in the shown in the red curve in Figure 3.1, corresponding to short-range magnetic order. Below the transition, in contrast, the spin correlations persist undamped throughout the lattice, constituting long-range magnetic order below the transition (blue mPDF in Figure 3.1). Even for a system as simple as a 20×20 2-D square grid of Ising spins modeled via Monte Carlo simulations, I see key features of real magnetic materials, including well-defined short-range magnetic correlations above the transition that grow into long-range order below the transition.

After generating the simulated mPDF patterns, I then fit a model of the ideal 2-D antiferromagnetic square lattice to the simulated patterns, where I refined a correlation length and a scale factor

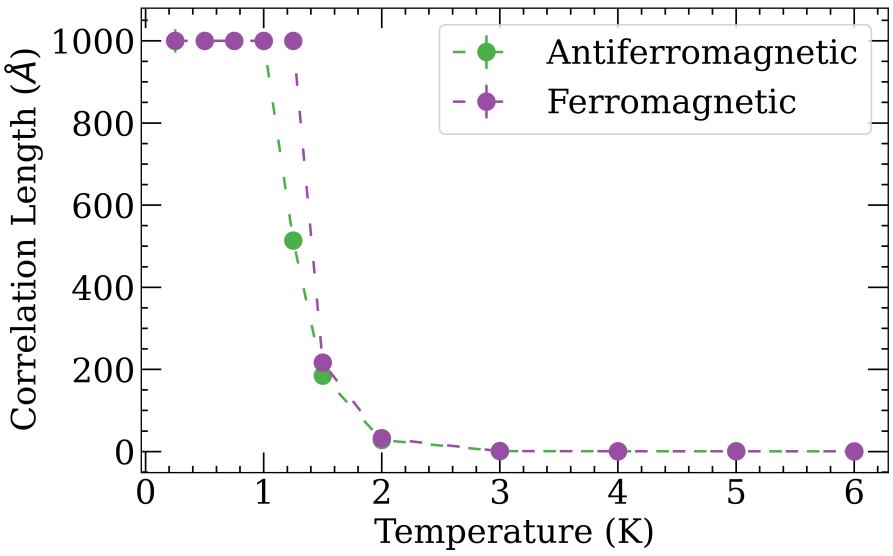

Figure 3.2 View of the correlation length for The 2-D Ising model Monte Carlo simulations. Both the ferromagnetic and antiferromagnetic correlation lengths are shown here. There is a dramatic increase around the transition temperature, but it is hard to see exactly where that is due to the amount the correlation length blows up.

corresponding to the ordered moment between nearest-neighbor spins. The best-fit correlation length versus temperature is shown over a wide temperature range in Figure 3.2. I see a rapid increase in correlation length as the temperature approaches the transition temperature 2.269 K from above. I note that since I am using a 20×20 Å^2 size simulation, any correlation length longer than about three times the spatial dimension of the simulation is functionally infinite. With decreasing temperature, the correlation length quickly passes into this regime, making it effectively infinite. A more zoomed-in look at the correlation length focused around the transition region 2-5 K gives more information about the critical behavior, shown in Figure 3.3 (green symbols) along with the ordered moment versus temperature (purple symbols). I see the correlation length increase slightly as the temperature decreases from 5 K to about 2.5 K, below which there is a sharp increase, and the correlation length rapidly increases to an effectively infinite value, which suggests the magnetic transition for this simulation happens near this temperature, which matches closely with the theoretical 2-D Ising transition around 2.269 K. The ordered magnetic moment

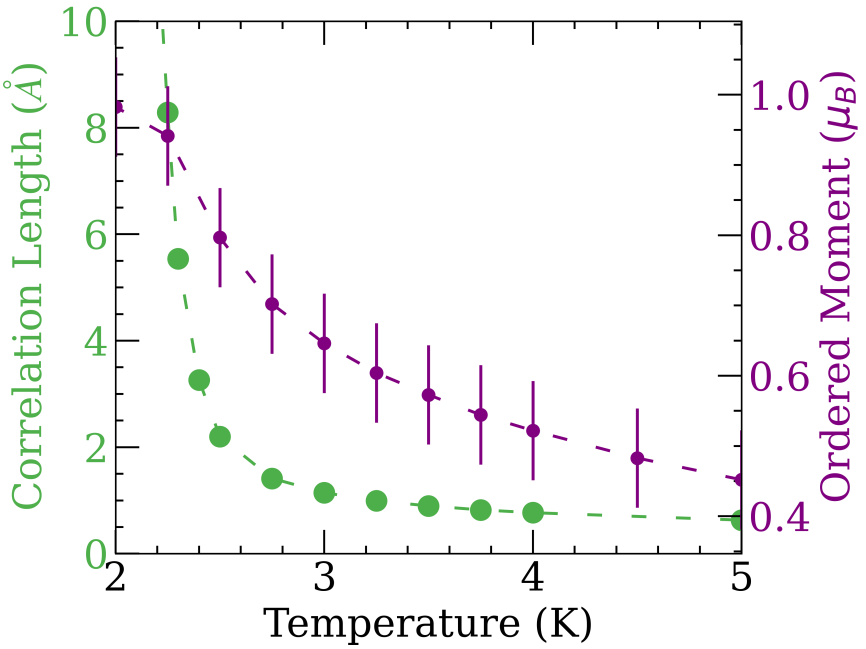

Figure 3.3 The green with the axis on the left outlines a zoomed in view of the correlation length for the antiferromagnetic mPDF focused around the magnetic transition. This highlights the increase that experienced approaching the magnetic transition, where short-range correlations persist, and confirm the magnetic transition is at the expected temperature. The purple shows the calculated local ordered moment for the Ising model as a function of temperature. The ordered moment is scaled to one at low temperature, as the method of calculating the ordered moment is not scaled to expected values.

determined from fitting the simulated mPDF also shows a steady increase as the temperature is lowered toward the transition before then increasing much more sharply below the transition. I note that the ordered moment in Figure 3.3 is defined as the correlated moment between nearest-neighbor spins, so it can remain nonzero above the transition. In contrast, the ordered magnetic moment averaged over the entire system (not shown here) becomes nonzero only at the transition point. The persistence of the locally ordered magnetic moment above the transition again reveals the presence of short-range correlations in our simulations. In fact, even up to 5 K, a temperature far above the magnetic exchange interaction energy of $J = 1$ K, the locally ordered moment is only just under half the low-temperature ordered moment. Short range correlations persist to very high temperatures at very significant values.

On a technical note, I mention that I scaled the calculated ordered moment to be 1 μ_B at low temperature, corresponding to full ordering of the $S = 1/2$ spins, which is verified directly from the fully-ordered spin configurations produced by the Monte Carlo simulations at low temperature. This is necessary due to the specifics of how the calculated mPDF is scaled in diffpy.mpdf.

3.2 Application to a Real Material: Antiferromagnetic MnTe

Having verified that mPDF modeling of Monte Carlo simulations can yield useful information about the magnetic correlations and transition in the 2-D Ising model, I performed the same procedure with MnTe to explore the value of this approach for gaining a deeper understanding of real materials. I loaded a .cif with the known hexagonal crystal structure of MnTe into my Monte Carlo code and simulated the mPDF using published values of the four nearest-neighbor exchange interactions [55], which are all the nearest-neighbor interactions needed to simulate the magnetic structure of MnTe. I created a supercell of 5 unit cells, making each spin unique out to about 15 Å from a given reference atom, so the simulated mPDF can only be calculated to 15 Å before encountering artifacts from

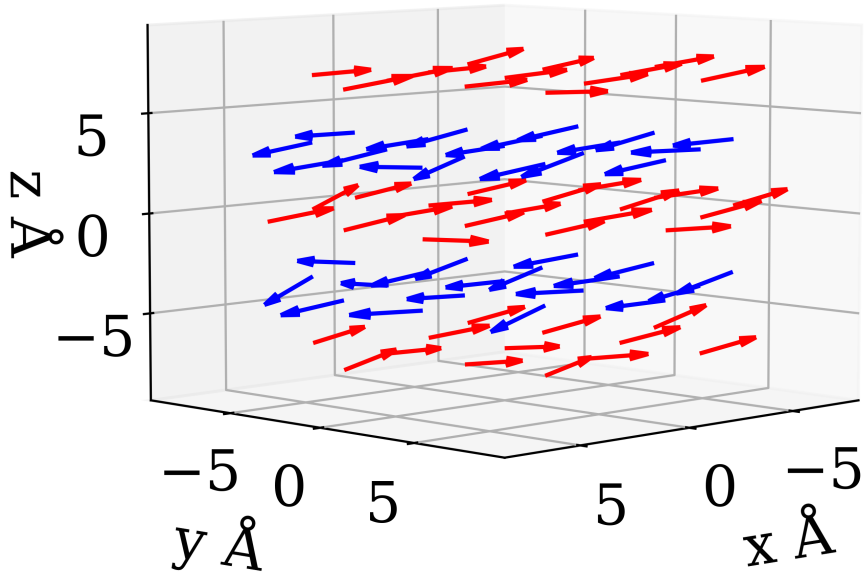

Figure 3.4 Resultant Monte Carlo spin configuration of MnTe at T=2 K. The red spin are pointing net in the negative x direction, while blue the net positive x direction. The spin configuration shown is one possible resultant configuration of many different possible non-unique solutions. This particular result showcases the spins falling into the correct planes. Other solutions will feature the spins falling into a plane, not necessarily the correct plane for the anisotropy in the material.

duplicated spins. 15 Å is a comfortable size to see the applicable features from the mPDF for MnTe without creating a lattice too large for a comfortable computation time.

As a first check, I verified that the low-temperature spin configuration achieved by the Monte Carlo simulations corresponds to the experimentally determined magnetic order in MnTe, with spins aligned ferromagnetically in the *ab* plane and modulated antiferromagnetically along the *c* axis. One such configuration reached by a Monte Carlo simulation is shown in Figure 3.4. While this example happens to show the spins oriented approximately within the planes (the known spin configuration for MnTe–see Figure 1.2) the simulations are not sensitive to the absolute orientation of the spins, and so other simulations fall into whatever random direction happens to dominate when sequentially lowering the temperature. One option is to include a magnetic anisotropy term in the Monte Carlo energy calculation, but for reasonable values of the anisotropy energy, I found

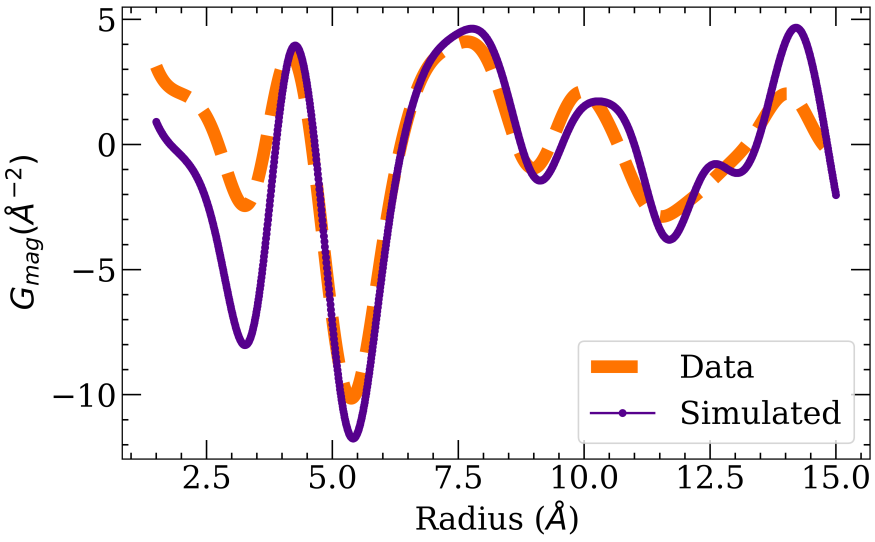

Figure 3.5 Comparison between simulated and experimental unnormalized mPDF data, with the data in orange and the simulation in purple. The x-axis gives different distances r from an arbitrary reference atom, while the y-axis gives the unnormalized mangetic correlation at that r.

that the impulse toward randomness in the Monte Carlo simulations prevents the spins from falling into the preferred orientation, most likely due to its small size when compared to the exchange parameters. The Monte Carlo simulations I built do not have the weight needed in the Monte Carlo simulations to overcome whatever random orientation it is in. The most important feature of MnTe, the ferromagnetic planes with antiferromagnetic alignment between adjacent planes, is reliably obtained. Although the absolute orientation of the spins does impact the calculated mPDF to some extent, the major features of the mPDF rely predominantly on the relative orientations, which my simulations reproduce accurately.

An example of a simulated mPDF pattern compared to experimental data for MnTe is shown in Figure 3.5 at 320 K, which is slightly above the magnetic transition ($T_C = 307$ K) and in the correlated paramagnetic regime where short-range correlations dominate. To compare more directly to the experimental data, I use the “unnormalized” mPDF, which has not been deconvoluted from the effect of the magnetic form factor. I see very good agreement between the simulated and the

experimental data, providing important confirmation that my Monte Carlo simulations can provide a good model of the short-range magnetism observed above T_N in MnTe. The exception to this observation is the first peak, where the peak in the simulation is significantly deeper than it should be. This is due to my simulation's lack of sensitivity to the correct absolute orientation. The alignment of the spins within the *ab* plane in the real material causes the nearest-neighbor peak to be shifted systematically higher along the *y*-axis than the simulation, as seen in Figure 3.5. my simulations reproduce the parallel alignment of the spins in common *ab* planes, but the absolute orientation of the spins is selected at random, causing the disagreement with the data.

To reveal the temperature evolution of the local magnetic correlations in MnTe, I show in Figure 3.6 a selection of the resultant average mPDF patterns for different temperatures between 236 K and 590 K. Here, I display the "normalized" mPDF, which is much sharper in real space than the unnormalized mPDF. Far above the transition temperature, I see only a few small peaks at positions corresponding to the first few near neighbors, reflecting the short-range order present. Although the spatial extent of the correlations is very short at these temperatures, I find it notable that well-defined correlations nevertheless persist over the first few neighbors at such high temperatures. The short-range correlations grow as the temperature is lowered toward the magnetic transition, as seen by peaks appearing at higher r values corresponding to distinct Mn-Mn interatomic distances in the atomic lattice of MnTe. For the two lowest temperatures shown in the plot (295 K and 236 K), the mPDF peaks persist over the full range with no damping at higher distances, indicating that long-range magnetic order has been reached in the simulation. The larger amplitude at 236 K compared to 295 K reflects the growth of the ordered moment (i.e., the non-fluctuating portion of the local moment that participates in the long-range magnetic order) as the temperature decreases.

I extracted the best-fit correlation length and locally ordered moment by fitting a model of the known magnetic structure to the mPDFs generated by the Monte Carlo simulations, just as I did for the 2-D Ising model. In this case, I have existing experimental data for comparison, first published

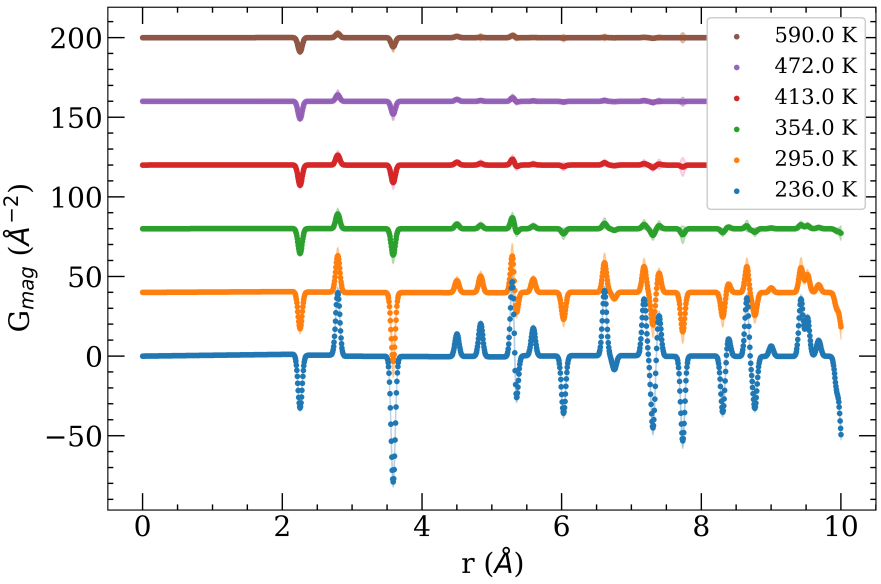

Figure 3.6 A sampling of the simulated mPDF for MnTe at different temperatures. The listed temperature in the legend is the scaled temperature for each mPDF. Each temperature occurs at the same zero point, but each are offset by 40 $Å^{-2}$ from each other for ease of viewing for the reader.

by Baral *et al*. [52], as shown in Figure 3.7. As my Monte Carlo simulations are finite in size and only account for the magnetic exchange parameters (not the anisotropy energy), one might expect a slight discrepancy in the transition temperature found by the simulations and observed in real life. Other papers featuring Monte Carlo simulations have encountered the same problem for similar reasons [60].Indeed, I found that the simulations predicted a transition temperature around 260 K compared to 307 K in the real material. The magnetic transition is a relatively easy parameter to extract for both the data and the simulations, as it is the point where the local ordered moment starts increasing dramatically. To provide a more meaningful comparison between the simulation results and the experimental results, I scaled the temperature points of the simulations such that the transition temperature matched observation with a multiplicative factor of 1.18, which brings 260 K to 307 K. The simulated temperatures shown in Fig. 3.7 are the scaled temperatures. The simulated ordered moment is also scaled down so that it agrees with the experimentally determined value of

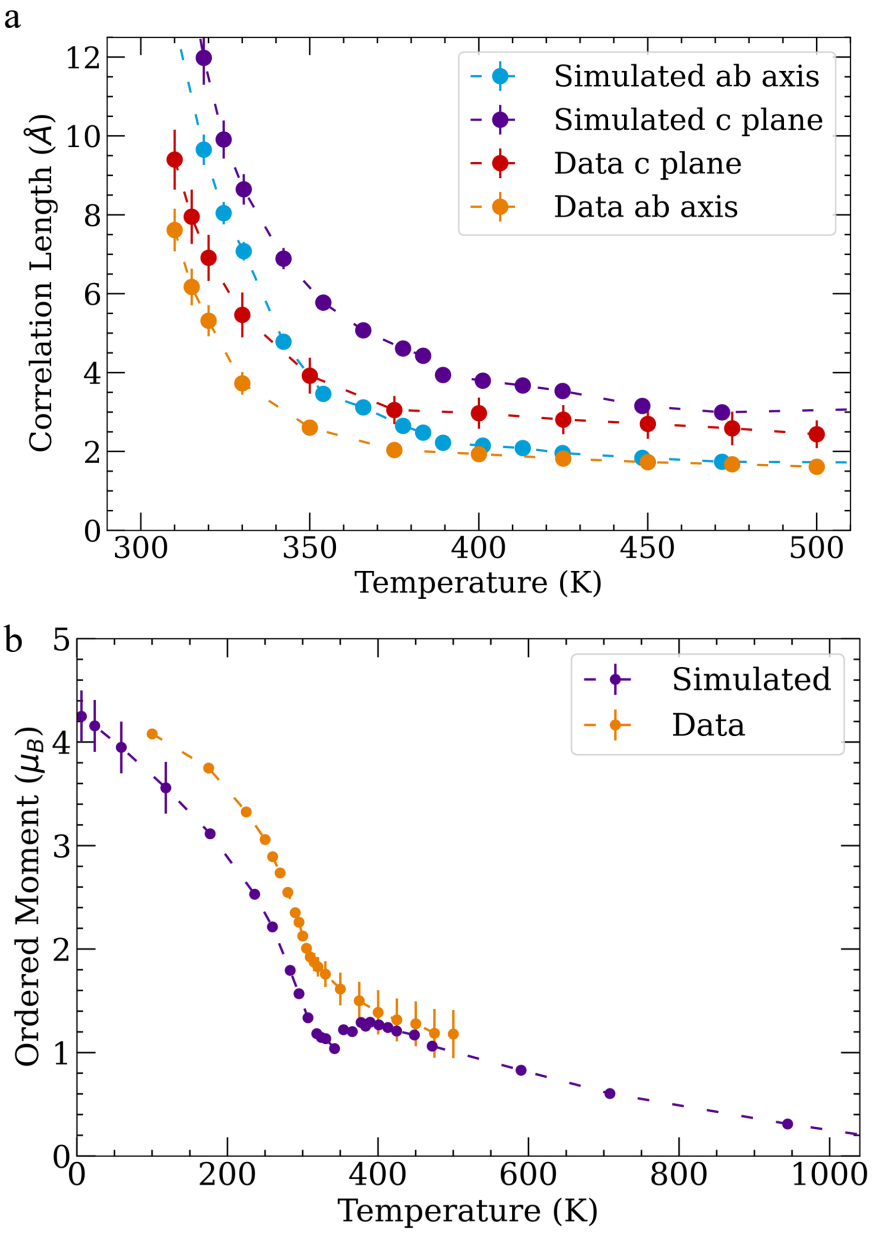

Figure 3.7 a) Simulated and experimental fits for the correlation length within the *ab* plane and along the *c* axis of MnTe as a function of temperature. b) Simulated and experimental fits for the local ordered moment of MnTe as function of temperature.

4.2 μ_B at low temperature [61]. This value is reduced from the maximum value of 5 μ_B expected for $S = 5/2$ spins due to partial delocalization of the Mn moment, something not accounted for in the Monte Carlo simulations themselves.

As seen in the figure, good agreement exists between the simulations and observations for both the correlation lengths and ordered moment, particularly in the limit of high temperature. I point out specifically the enhanced correlation length along the *c* axis relative to the *ab* plane, which is clearly seen in both the simulations and experiment. This is due to the much larger out-of-plane exchange interactions J_1 and J_3 compared to the in-plane interaction J_2. The fact that a subtle but meaningful feature such as this anisotropic correlation length can be discerned from the simulations is a positive indicator for the value of using Monte Carlo simulations to model mPDF data. I also note there is general agreement on the increase of the correlation length as the temperature approaches the magnetic transition from above. Both the simulation and the data show a rapid increase in correlation length beginning around 375 K.

The ordered moment likewise shows generally good agreement between the experimental and simulated data. The exception is a small decrease in the locally ordered moment as the temperature is lowered just above the transition, which ends up offsetting the low temperature ordered moment slightly. The cause of this slight downturn is currently unknown, but the overall agreement between the simulations and experimental observations remains.

Chapter 4

Results: Extracting Exchange Parameters from Monte Carlo Simulations

4.1 Examining the Parameter Space

Having confirmed the viability of using Monte Carlo simulations to model mPDF data with realistic exchange interactions, I next investigate the possibility of determining the exchange interactions from the Monte Carlo modeling directly, such that the technique could be applied without detailed prior knowledge of the exchange interactions. To do this, I ran Monte Carlo simulations for many different sets of exchange parameters that, by design, deviated from the published values. By comparing the measured data to the outputs of the simulations with different sets of exchange parameters, I can then identify the best set(s) of exchange parameters and evaluate how well the mPDF data constrain the magnetic interactions, essentially allowing us to explore the space of magnetic exchange interactions with reference to the mPDF data.

The resultant mPDF patterns at 320 K shown in Figure 4.1 are obtained by inputting the published J values scaled uniformly by varying amounts. It is clear qualitatively that the mPDF

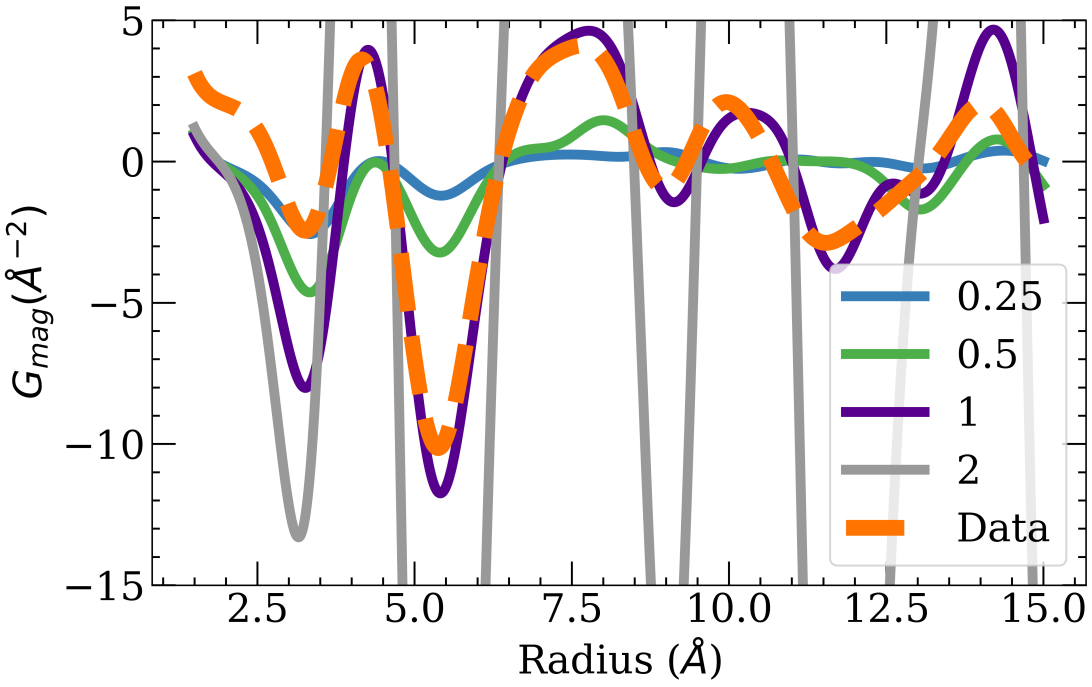

Figure 4.1 Simulated mPDF for different magnetic exchange interactions J, scaled uniformly from the published values for MnTe, compared to experimental data for the same material. The amount each is scaled is given in the legend. All simluations/data are at T = 320 K.

pattern using the actual published exchange interactions, given by the purple simulated mPDF (scaled by a factor of 1) in Figure 4.1, has the best agreement with the data. The great agreement between the data and simulation with the published values proves that Monte Carlo should at least be able to predict the magnetic exchange interactions to within a factor of order unity. I emphasize that this prediction is made at temperatures above the magnetic transition, where there are only short-range correlations.

One noticeable trend is that the simulations consistently overshoot the first nearest neighbor peak relative to the data, even with significantly smaller exchange parameters. I attribute the inaccurate first peak mainly to the fact that the spins in the real material orient within the *ab* plane, such that the interatomic vector connecting a spin to its nearest neighbor is perpendicular to the spins, since the nearest neighbors lie along the *c* axis. The mPDF contribution from such a spin pair consists of a line with a positive slope from the origin to the pair separation distance, followed by a negative peak reflecting the antiferromagnetic correlations. This situation can be seen in Fig. 3.1 for the 2-D Ising model, as both the simulated mPDF below and above the transition experience that upward slope due before the first nearest neighbor peak. When the mPDF pattern is averaged over all possible orientations of the pair of spins relative to the vector connecting them, the slope of the linear term in Equation 2.1 averages to zero. This is the case in my Monte Carlo simulations. As such, the nearest-neighbor peak in the experimental mPDF seen in Figure 4.1 (the orange graph) starts at a higher position along the *y* axis than the simulated peak does, making the Monte Carlo modeling of the nearest neighbor inaccurate. Even accounting for this difference in starting height, the simulated peak (shown in purple in Figure 4.1) tends to have a larger magnitude than the observed peak, which may point to some real effect in MnTe that slightly reduces the nearest-neighbor correlations.

To probe the exchange interaction parameter space more systematically, I created a grid of different J_1, J_2, and J_3 values and ran different Monte Carlo simulations with each of them, comparing the simulation results to the experimental data. As before, I used $T = 320$ K. I chose to omit J_4 in

these simulations, as it is the weakest of the magnetic interactions, and in my estimation, it does not impact the system enough to justify expanding the parameter space by another dimension. After calculating the mPDF, I then compared each resultant simulated mPDF to the experimental data, taking the sum of the squared difference between the simulation and data to quantify the residue. The different residue values shown in Figure 4.2 interpolated between $5 \times 5 \times 5$ values of J_1, J_2, and J_3 spanning a broad range. This coarse-grained approach to exploring the interaction space would be a reasonable starting point in cases where prior knowledge of the exchange parameters is not available. I scaled the residue down by a factor of 1000 for easier viewing in this figure. In the very middle of the set of graphs, where $J_1 = -47.98$ K, I find the minimum residue, corresponding to the experimentally determined values for MnTe ($J_1 = -47.98$ K, $J_2 = -0.29$ K, $J_3 = -6.382$ K). The residue far outside the known J values quickly becomes very large. The lowest overall residue is around 3280, while the points around it rise to 40,000 and beyond very quickly. These results demonstrate that this approach can be used to explore the parameter space broadly and identify a potential region of interest for a more fine-grained investigation.

I then generated another $5 \times 5 \times 5$ grid in the vicinity of the region identified by my coarse-grained exploration above, shown in Figure 4.3. I find a sort of cavity in parameter space around the published J values. The residue landscape in this region of parameter space is a somewhat shallow, and the absolute minimum is not right at the published values, but rather at $J_1 = -57.9865$ K, $J_2 = 9.7099$ K, and $J_3 = -1.3827$. All the combinations of exchange parameters that had lower residues than the published values are shown in Table 4.1, with the published values ranking as the 6th lowest set of values in this set of simulations. Part of the reason for this may be due to the lack of a J_4 in these simulations, but it is more likely that the parameter space inside of this cavity this region is just extremely shallow and hard for Monte Carlo simulations to navigate.

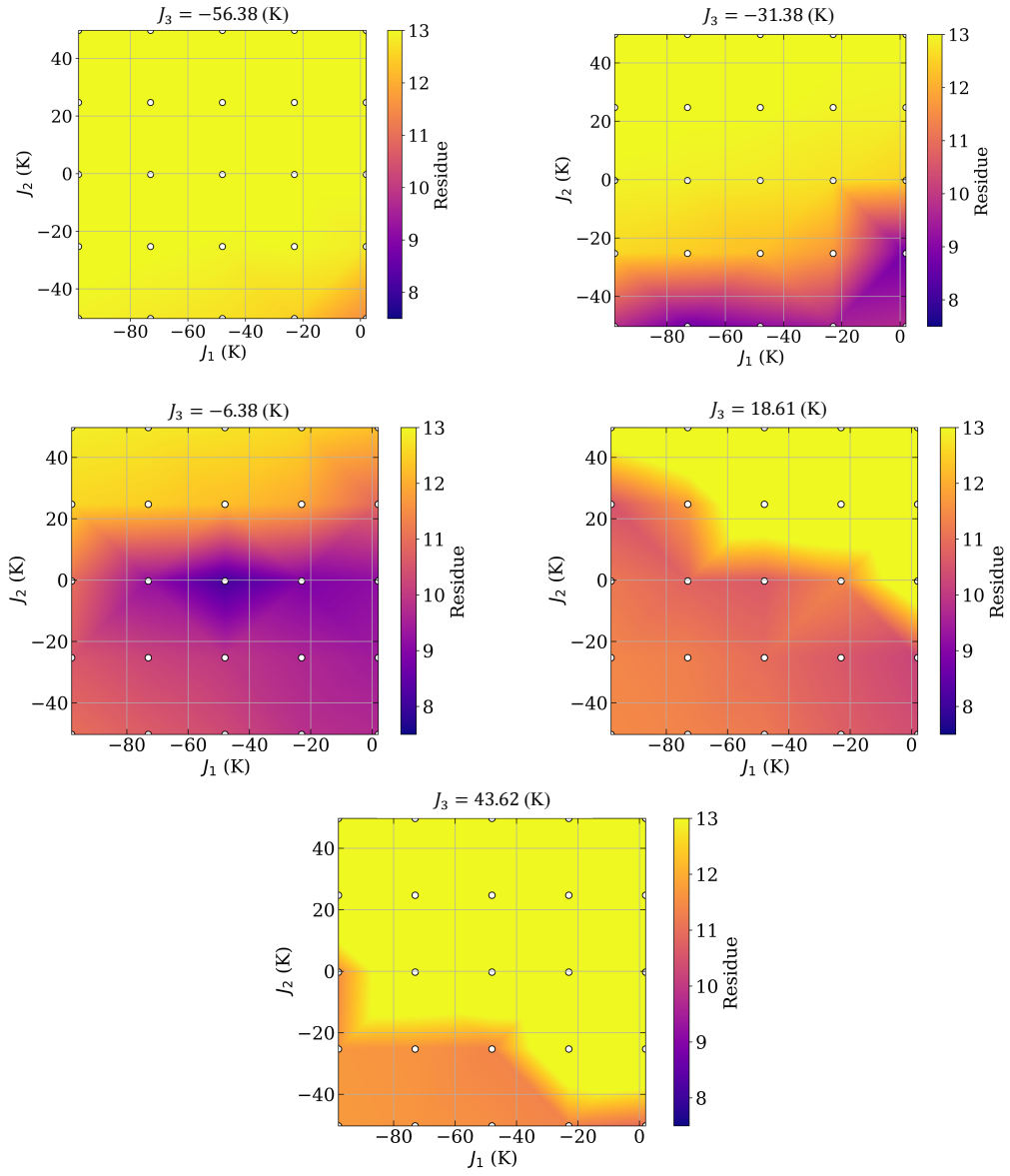

Figure 4.2 Residues from Monte Carlo modeling of the experimental mPDF for MnTe at 320 K for different values of J_1, J_2, and J_3, scaled logarithmically. Each slice is a different J_3 value, listed to the side of the graph, with an interpolated colormap of different residue values for different J_1 and J_2 values within that J_1 slice. The darker the color the lower the residue is.

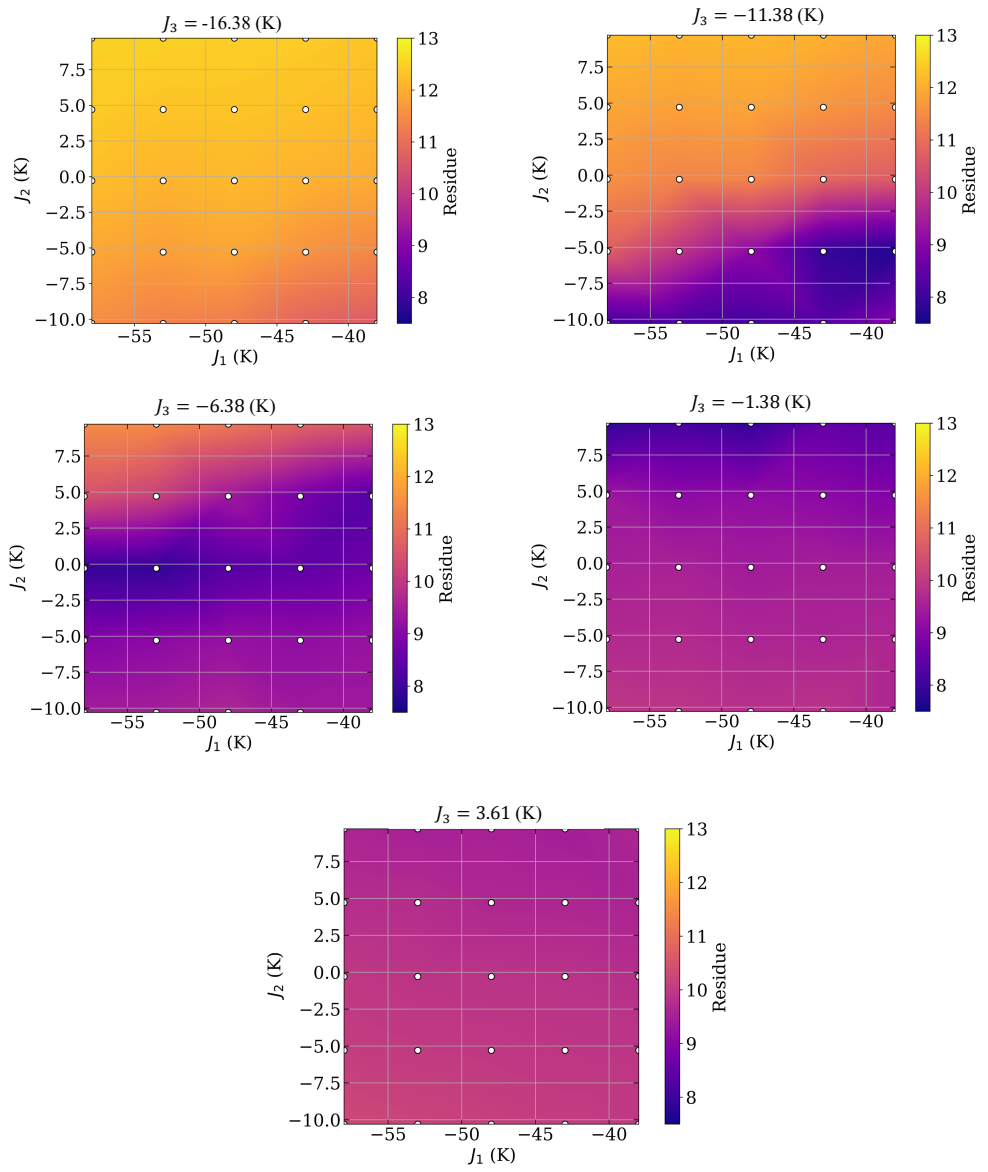

Figure 4.3 Residues from Monte Carlo modeling of the experimental mPDF for MnTe at 320 K for different values of J_1, J_2, and J_3, scaled logarithmically, in a tighter range than Figure 4.2. Each slice is a different J_3 value, listed as the title of the graph, with an interpolated colormap of different residue values for different J_2 and J_3 values within that J_1 slice. The darker the color the lower the residue is.

J_1 (K)	J_2 (K)	J_3 (K)	Residue
-57.9865	9.7099	-1.3827	2697.6541
-47.9865	9.7099	-1.3827	2713.2213
-47.9865	-10.2901	-11.3827	3287.1531
-52.9865	9.7099	-1.3827	3367.5248
-37.9865	4.7099	-6.3827	4095.1157
-47.9865	**-0.2901**	**-6.3827**	**4174.3484**
-42.9865	-10.2901	-11.3827	4197.2289

Table 4.1 Selected values for the residues shown in Figure 4.3, sorted by order of increasing residue, where the Residue= $\sum_i (y_i - \hat{y}_i)^2$, where y_i is the experimental G_{mag} and $\hat{y}_i$ the simulated G_{mag} for T = 320 K and the given exchange parameters J_1, J_2, and J_3. This shows the exchange parameters that resulted in lower residues than the published value. The bolded text shows the literature values.

While this parameter space may be rugged and hard to navigate, all of these values are relatively close to the published magnetic exchange parameters, demonstrating that my Monte Carlo modeling approach can recover the exchange parameters with reasonable success.

4.2 Predicting Exchange Parameters

Systematically exploring the space of exchange interactions through a pre-defined grid of exchange values can be effective, as I just showed, but it is also worth investigating the possibility of running an optimization procedure to find the best combination of exchange parameter. To be able to compare data and simulations directly, it is best to place the data and simulations on the same scale of absolute units, rather than the arbitrarily scaled mPDF data that normally comes off the beamline. Fortunately, the data can be scaled to absolute units simply by dividing by the overall scale factor determined from atomic PDF fits to the data and adjusting for the known ordered moment for MnTe.

To ensure the simulation was also on the correct scale, I used the `calculate_ordered_scale` function in diffpy.mpdf to obtain the value that scales the generated mPDF simulations to the same units of the data. Then, the calculated average mPDF is just scaled by this number.

I first tried nonlinear least-squares optimization to predict the exchange parameters. I implemented a scipy least-squares function, with a standard residual function. However, I found this approach to be unsuccessful, failing even to predict that the first nearest neighbor is antiferromagnetic, something that is qualitatively obvious just from looking at the correlations in the data. I believe this is due to the rugged nature of the parameter space, causing the least-squares algorithm to get trapped in local minima and terminate early.

To traverse this rugged complex parameter space more effectively, I turned to particle swarm optimization (PSO), a more robust method to traverse parameter space [62]. PSO generates a series of theoretical particles within the parameter space, that move along it with an attempt to fall into a minimum. Each particle has their own inertia (the initial value is standard across all particles in the algorithm) that changes depending on the steepness of the particle space they are in. Each particle also has an ability to talk to the swarm, which the user limits based on how much communication they think is needed in the space to avoid falling into local minima. The main drawback of particle swarm optimization is how computationally expensive it is.

A single iteration of PSO is as follows. The average mPDF needs is calculated calculated for each particle (I used fifteen particles). I ran forty full Monte Carlo simulations per particle, and then calculated the average mPDF at each particle site. Next I calculated the reside at each particle, and PSO updated the inertia of the particle based on what was found at the particle of interest and what was found at the other nine particles in the simulation. Each particle then moved a short distance through the parameter space finding a new set of exchange constants to use for the next iteration. Since my Monte Carlo simulations are already fairly computationally expensive on

their own, computing even four hundred for one iteration of PSO makes PSO a computationally expensive task.

For a first look at PSO, while attempting to avoid excessive computation times for this initial exploration, I ran PSO with 30 iterations, with each calculated mPDF resulting from the average of 40 Monte Carlo simulations, along with the published mPDF data of various different temperatures, shown in Table 4.2. At a first glace, it seems the PSO is comparable to the least squares results. However, the best cost values are quite high, especially those at lower temperatures. I conjectured this may be because of the low number of iterations used. I picked just one temperature, the 320 K data that I have been using the most throughout this project, for a much longer simulation.

Temperature (K)	J_1 (K)	J_2 (K)	J_3 (K)	J_4 (K)	Best Cost
310	-18.2048	-9.7223	-16.3897	-8.2121	3.51e+3
315	-5.3319	4.4835	-7.1545	7.9858	1.31e+3
320	6.0994	-2.2630	-0.4050	9.3556	156
330	-16.4465	3.3336	-5.2570	8.8997	663
350	-10.0009	4.3665	-3.7420	12.1707	443
375	-12.2857	4.3250	-2.2160	4.0204	230
400	-10.4876	0.1529	-2.6611	-5.5393	196
425	-11.5528	-0.1286	-0.6172	4.8680	140
450	-1.5726	0.9923	-1.7473	10.7189	120

Table 4.2 Resultant exchange interaction parameters at various temperatures after running 30 iterations of PSO, and 40 Monte Carlo simulations contributing to the average mPDF of each particle in each iteration.

I ran 1000 iterations of PSO on the 320 K data. A larger amount of simulations was more successful than my small 30 iteration runs. I found these values: $J_1 = -57.965$ K, $J_2 = 11.959$ K, $J_3 = 0.95627594$ K, $J_4 = -5.23782187$ K, with a best cost of 6.74e-5. While not exactly the published

values, these are in the correct vicinity, with the exception of J_2. I find this very promising for the future of fitting exchange parameters from PSO. However, this simulation took over a week to run on its own on a standard desktop, so efforts should be made to make it more efficient.

Chapter 5

Discussion

I have successfully used Monte Carlo simulations to simulate the mPDF of a material from given exchange parameters and crystal structure. I have shown my simulations to be successful on a 2-D Ising model and for the antiferromagnetic semiconductor MnTe. The software infrastructure I have developed can be applied to any magnetic material for which the positions of the magnetic atoms are known and a .cif file is available.

Overall, the simulations for MnTe match the experimental data very well. I emphasize the novelty of this approach–instead of refining the spin configurations directly, as is conventionally done in mPDF analysis, I showed that I can accurately reproduce the short-range magnetic correlations with just the exchange parameters. I have also demonstrated the potential to be able to work in the opposite direction in the future, i.e., start with the data and refine the interaction parameters from the simulations.

This development provides an important alternative to the conventional method of determining magnetic exchange parameters that relies on low-temperature inelastic neutron scattering data collected when the material is in a fully ordered magnetic state. Some of the most notable materials in recent condensed matter research, like geometrically frustrated magnets [63] quantum spin liquids [28], and spin ices [64, 65], are dominated only by short-range correlations with no long-

range order. In such situations, conventional approaches can only be taken by applying an extremely large magnetic field to polarize the system into a uniform ferromagnetic state, but such a large field may change the underlying interactions in the first place. Doing Monte Carlo simulations on higher-temperature mPDF data provides another option, in addition to Spinteract [31], to use short-range correlations to estimate exchange parameters for materials.

In addition, it is not always feasible to perform inelastic neutron scattering experiments at the low temperatures required. Cooling materials to near zero kelvin is difficult and expensive. mPDF can be taken at higher temperatures above the transition and still give useful information about the magnetic exchange constants. Some materials are also difficult to synthesize as single crystals. The mPDF is usually obtained with powdered crystals. Monte Carlo mPDF simulations offer a simpler alternative in order to find the magnetic exchange interaction from a powder instead of a large single crystal.

The main problem with the Monte Carlo simulations that I implemented seems to be that a couple of important factors were omitted from the model. This causes the transition temperature of the simulation to disagree with the observed transition temperature, so any fits to simulated data need to be scaled in temperature to match the known transition temperature. To extract meaningful comparisons between the simulations and experimental data for the locally ordered magnetic moment and magnetic correlation length, such as those shown in Figure 3.7, this temperature scaling must be done carefully. Improving my Monte Carlo simulations to capture the transition temperature more accurately would be a valuable development. Nevertheless, fits to the experimental and simulated data exhibit important qualitative agreement, such as the shape of the gentle increase in correlation length as the temperature is lowered shown in Fig. 3.7a. While they do need to be scaled, the existence of these short-range correlations in Monte Carlo shows that they can be modeled from only the exchange parameters. Consequentially, I should be able to extract these

exchange values from details about the short-range correlations, if the transition temperature is known for proper scaling.

The same outlook is provided by the simulated locally ordered moment shown in Figure 3.7b. There appears to be an artifact right around the transition temperature, where there is a dip in the locally ordered moment before it begins increasing at a higher rate at the magnetic transition. The origin of this behavior in the simulations is unclear, but it is possible that a finite cell size is causing some sort of artifact in the magnetic lattice, or the nature of Monte Carlo itself could be the cause. Monte Carlo has some difficulties right around transitions as the two regimes of order and disorder compete and cross from short-range correlations to long-range order.

Magnetic anisotropy energy, by which the energy of the system is lowered when the spins are oriented along particular directions with respesct to the crystal lattice, was also omitted from my simulations. However, this is not a fundamental limitation of the approach, as there have been successful Monte Carlo simulations that incorporate anisotropy [66]. The authors of that study also use the Metropolis algorithm, but modify their Hamiltonian to be more complex than just the single added anisotropy term that I included in my simulations. A deeper look into anisotropic Monte Carlo simulations could solve many of the problems seen in my current simulations, most notably the lack of the anisotropic features in the low-r regime that seemed to interfere with the accuracy of my Monte Carlo modeling. Correcting this issue could also simplify the complex parameter space found around the true J values around MnTe.

Despite the limitations of the current study, I find it promising that the simulations nevertheless identified the correct range of exchange values within a factor of order unity. Even without prior knowledge, one would be able to use this approach to make at least an approximate prediction of the exchange parameters, though finding the exact values may be difficult due to the rugged nature of the local parameter space.

My implementation of particle swarm optimization was promising, if ultimately not completely successful. However, there may be ample opportunity to improve the implementation of this optimization method as applied to mPDF data and magnetic exchange interactions. Each particle has its own inertia and ability to talk with different points within the parameter space about what minima it has found. The algorithm allows different values of the parameters that control the inertia and communication tendencies of the particles, and different types of data and fitting landscapes may have different optimal combinations of these values. The process of choosing which values are best is complicated. Designing a better set of parameters for particle swarm optimization could potentially enable it to explore the parameter space more effectively. I used the default values as implemented in the python package pyswarm for a spherical parameter space, which took about 6-8 hours to do just 30 iterations, when optimally around 100 simulations should be the minimum amount of iterations used for particle swarm optimization. This aspect of my analysis can likely be significantly improved.

I also emphasize that even in the absence of a reliable and robust optimization routine to fit the exchange parameters to the data, significant insights can still be gained by Monte Carlo simulations of mPDF data. For many materials, there are competing theoretical predictions of the Hamiltonian, including the nature and strength of the exchange interactions, with important implications for the physics of the material depending on which model Hamiltonian best describes the material. In such situations, I envision this approach as being a valuable method for testing competing models and narrowing down parameter space, especially when conventional inelastic neutron scattering is not possible.

Chapter 6

Summary and Conclusion

In this work, we have shown that Metropolis Monte Carlo simulations of mPDF data are a viable method to investigate magnetic exchange parameters in real materials, especially for those materials that do not fully order at accessible temperatures (or at all). Short-range correlations are driven by the underlying magnetic interactions, and so modeling the mPDF signatures of the short-range correlations can yield rich information about those interactions and the physics of the material. Indeed, short-range correlations are observed in many magnetic materials, and some materials are even dominated completely by them, lacking any long range order [14–17]. Understanding short-range magnetic correlations increases our understanding of many technologically important materials and quantum materials. Since these correlations are most naturally described in real space rather than momentum space, modeling the real-space mPDF through Monte Carlo simulations provides a natural way to connect the data to the relevant interactions.

We successfully demonstrated the utility of this approach for a simple 2-D Ising model with nearest-neighbor interactions. The Monte Carlo simulations accurately predicted the transition temperature for the theoretical Ising modes, as well as the temperature dependence of the local order parameter and correlation length, showing the increasing short range correlations in the lattice as the temperature is lowered toward the transition temperature at 2.269 K.

We also benchmarked this approach using a real material, antiferromagnetic MnTe. We showed that the simulations agree with existing mPDF data and capture the temperature evolution of the short-range magnetic correlations above T_N reasonably well. We see good agreement between data and our simulations for both the correlation length and ordered moment, confirming that short range correlations can be modeled just from the exchange parameters using Monte Carlo simulations.

Robustly extracting exchange parameters from data using optimization routines will require further research. We can confidently predict the general order of magnitude of the exchange parameters, but the actual parameter space around the expected values can be quite rugged. We are confident that with further research and finding a better way to characterize and explore the space, the use of Monte Carlo simulations can be further developed to provide a reliable way to determine the exchange parameters of materials. This will enable deeper understanding of important magnetic materials with significance for both fundamental and applied research.

Appendix A

Monte Carlo Code

We present the code used to simulate the mPDF for materials.

```
import numpy as np
import random
from diffpy.mpdf import *
from diffpy.structure import loadStructure
import sys
import pickle
import statistics
import math
import pandas as pd


def calcEnergy(mstruc, NN1, NN2, NN3, NN4, J1, J2, J3, J4):
    """
    mstruc = MagStructure object
    index = index of spin to be considered S0
```

```
    NN1,NN2,NN3 = dictionary where each key is an index of a spin in the
                original cell (before the tiling for periodic boundary
                conditions) and the corresponding value in the dictionary
                is a list of indices corresponding to the nearest-neighbor
                spins for first, second, and third nearest-neigbors
    J1,J2,J3 = different interaction energy values for the respective
                nearest-neighbor distance
    """
    energy = 0
    for atoms in mstruc.calcIdxs:
        S0 = mstruc.spins[atoms]
        Sj1 = mstruc.spins[NN1[atoms]]
        for spin in Sj1:
            energy -= J1 * np.dot(S0,spin)

        Sj2 = mstruc.spins[NN2[atoms]]
        for spin in Sj2:
            energy -= J2 * np.dot(S0,spin)

        Sj3 = mstruc.spins[NN3[atoms]]
        for spin in Sj3:
            energy -= J3 * np.dot(S0,spin)

        Sj4 = mstruc.spins[NN4[atoms]]
        for spin in Sj4:
```

```
            energy -= J4 * np.dot(S0,spin)

        anisotropy = 1 #Kelvin
        for spin in mstruc.spins:
            energy -= anisotropy*spin[2]**2

    return energy/2

def deltaEnergy(mstruc, index, NN1, NN2, NN3, NN4, J1, J2, J3, J4,      q
    """
    mstruc = MagStructure object
    index = index of spin to be considered S0
    NN1,NN2,NN3 = values where each key is an index of a spin in the
                original cell (before the tiling for periodic boundary
                conditions) and the corresponding value in the dictionary
                is a list of indices corresponding to the nearest-neighbor
                spins for first, second, and third nearest-neigbors
    J1,J2,J3 = different interaction energy values for the respective
                nearest-neighbor distance
    newspin = new spin to evaluate
    """
    energy1 = 0
    energy2 = 0
    S0 = mstruc.spins[index]
```

```
Sj1 = mstruc.spins[NN1[index]]
for spin in Sj1:
    energy1 -= J1 * np.dot(S0,spin)

Sj2 = mstruc.spins[NN2[index]]
for spin in Sj2:
    energy1 -= J2 * np.dot(S0,spin)

Sj3 = mstruc.spins[NN3[index]]
for spin in Sj3:
    energy1 -= J3 * np.dot(S0,spin)

Sj4 = mstruc.spins[NN4[index]]
for spin in Sj4:
    energy1 -= J4 * np.dot(S0,spin)

anisotropy = 1 #Kelvin
energy1 -= anisotropy*S0[2]**2



Sj1 = mstruc.spins[NN1[index]]
for spin in Sj1:
    energy2 -= J1 * np.dot(newspin,spin)

Sj2 = mstruc.spins[NN2[index]]
```

```
    for spin in Sj2:
        energy2 -= J2 * np.dot(newspin,spin)

    Sj3 = mstruc.spins[NN3[index]]
    for spin in Sj3:
        energy2 -= J3 * np.dot(newspin,spin)

    Sj4 = mstruc.spins[NN4[index]]
    for spin in Sj4:
        energy2 -= J4 * np.dot(newspin,spin)

    energy2 -= anisotropy*newspin[2]**2

    return energy2 - energy1

# Function to perform a Monte Carlo step
def monte_carlo_step(mstruc, J1, J2, J3, J4, NN1, NN2, NN3, NN4,
        beta, copydict):
    """
    a singular monte carlo step: changes the spin and
            evaluates whether the change should be taken
    mstruc = MagStructure object
    index = index of spin to be considered S0
    NN1,NN2,NN3 = values where each key is an index of a spin in the
                original cell (before the tiling for periodic boundary
```

```
            conditions) and the corresponding value in the dictionary
            is a list of indices corresponding to the nearest-neighbor
            spins for first, second, and third nearest-neigbors
J1,J2,J3 = different interaction energy values for the respective
            nearest-neighbor distance
beta = value to plug into exponetial for monte carlo, should
        be in units of kb/T
copydict = a dictionary that links the duplicated supercells to
        each other in order to keep them identical them idential
        throughout simulations
"""
original_size = len(mstruc.calcIdxs)
# choose a spin at random
idx = random.randint(0, original_size - 1)
# generate a new random spin for a candidate move

theta = np.arccos(np.random.uniform(-1, 1))
phi = np.random.uniform(-np.pi, np.pi)
newspin = np.array([np.sin(theta)*np.cos(phi),
        np.sin(theta)*np.sin(phi), np.cos(theta)])
newspin *= np.linalg.norm(mstruc.spins[idx])
# calculate change in energy
delta_energy = deltaEnergy(mstruc, idx, NN1, NN2, NN3, NN4, J1, J2,
                           J3, J4, newspin)
if delta_energy < 0:
```

```
        mstruc.spins[idx] = 1.0*np.array(newspin)
        update_spins(mstruc, idx, copydict)
        #print('good move accepted')
    elif (delta_energy > 0) and (random.random() < np.exp(
            -beta * delta_energy)):
        mstruc.spins[idx] = 1.0*newspin
        update_spins(mstruc, idx, copydict)
        #print('bad move accepted')
    #else:
        #print('move rejected', delta_energy)

# Main Monte Carlo simulation
def monte_carlo_simulation(temperature, num_steps, mstruc,
        J1, J2, J3, J4, NN1, NN2, NN3, NN4, copydict):
    """
    runs a singular monte carlo simulation at a given temperature
    temperature = target temperature
    num_steps = number of Monte Carlo steps to take
    mstruc = MagStructure object
    index = index of spin to be considered S0
    NN1,NN2,NN3 = values where each key is an index of a spin in the
                original cell (before the tiling for periodic boundary
                conditions) and the corresponding value in the dictionary
                is a list of indices corresponding to the nearest-neighbor
                spins for first, second, and third nearest-neigbors
```

```
    J1,J2,J3 = different interaction energy values for the respective
                nearest-neighbor distance
    beta = value to plug into exponetial for monte carlo, should be in
            units of kb/T
    copydict = a dictionary that links the duplicated supercells to each
            other in order to keep them identical them idential throughout
            simulations
    """
    beta = 1.0 / temperature  # Inverse temperature

    for step in range(num_steps):
        monte_carlo_step(mstruc, J1, J2, J3, J4, NN1, NN2, NN3, NN4,
                             beta, copydict)

# Function to copy changes in the original cell to the extra cells to
        ensure periodic boundary conditions
def update_spins(mstruc, index, copydict):
    """
    updates the identical spins in a lattice
    mstruc = MagStructure object
    index = index of spin in the original cell which should be copied out
            to all the duplicate cells
    copydict = dictionary pairing each spin index in the original cell with
               the indices of all its duplicates
    """
```

```
    mstruc.spins[copydict[index]] = 1.0*mstruc.spins[index]



def reset_spins(mstruc, copydict):
    """
    resets the spins
    mstruc = MagStructure object
    copydict = a dictionary that links the duplicated supercells to each other in
               order to keep them identical them idential throughout simulations
    """
    original_size = len(mstruc.calcIdxs)
    sx = np.random.choice([-1, 1], size=original_size)
    sy = np.random.choice([-1, 1], size=original_size)
    sz = np.zeros_like(sx)

    mstruc.spins[:original_size] = np.transpose((sx, sy, sz))/np.sqrt(2)*5/2
    for idx in range(original_size):
        update_spins(mstruc, idx, copydict)



struc = loadStructure('MnTe_hexA.cif')



qdamp = 0.024
```

```
rmin = 0.1
rmax = 90

# set up the MagSpecies
mspec = MagSpecies(struc=struc,rmaxAtoms=rmax,ffparamkey='Mn2')
mspec.strucIdxs = [2, 3] # determined from previous inspection of the unit
    cell
mspec.basisvecs = np.array([1, -1, 0])/np.sqrt(2)*5/2 # any vector
        perpendicular to the c axis will be fine
#Divide root 2 multiply 5/2 to scale spins for MnTe
mspec.kvecs = np.array([0, 0, 1]) # this gives the desired AF structure
        for this choice of unit cell

# set up the MagStructure
mstruc = MagStructure(rmaxAtoms=rmax)
mstruc.loadSpecies(mspec)
mstruc.makeAll() ### populates the spin and atom arrays

# create the MPDFcalculator for MnTe
mc = MPDFcalculator(magstruc=mstruc,qdamp=qdamp,rmax=rmax,rmin=rmin)
lat = struc.lattice
fraccoords = lat.fractional(mstruc.atoms)
fracx, fracy, fracz = fraccoords.T
spinx, spiny, spinz = mstruc.spins.T
```

```
#create mask that limits the unit cell size
maskx= np.array([])
masky= np.array([])
maskz= np.array([])
maskSpinx= np.array([])
maskSpiny= np.array([])
maskSpinz= np.array([])

supercellx=5
supercelly=5
supercellz=5

#loop through cell and catch all the positions/spins of the desired atoms
for j in range(len(fracx)):
    if (-(supercellx/2-0.01) < fracx[j] < (supercellx/2+0.01) and
        -(supercelly/2-0.01) < fracy[j] < (supercelly/2+0.01) and
        -(supercellz/2-0.01) < fracz[j] < (supercellz/2+0.01)):
        maskx = np.append(maskx,fracx[j])
        masky = np.append(masky,fracy[j])
        maskz = np.append(maskz,fracz[j])
        maskSpinx = np.append(maskSpinx,spinx[j])
        maskSpiny = np.append(maskSpiny,spiny[j])
        maskSpinz = np.append(maskSpinz,spinz[j])

mask = np.array([maskx,masky,maskz]).T
```

```
maskSpin = np.array([maskSpinx,maskSpiny,maskSpinz]).T
#mask = lat.cartesian(mask)
print('Finished desired atoms')

#apply mask to atoms and spins
mstruc.atoms=mask
mstruc.spins=maskSpin

# create a copy of the original spins and atomic positions for later use
original_spins = 1.0*mstruc.spins
original_atoms = 1.0*mstruc.atoms

# set calcIdxs attribute to include every spin in the original cell
mstruc.calcIdxs = range(len(mstruc.atoms))

# choose a small enough Uiso value to be able to resolve distinct peaks
mstruc.Uiso = 0.002

#tile the initial spin configuration to create periodic boundary
    conditions
lst = [-1, 0, 1]
#print('starting boundry conditions')
for i in lst:
    for j in lst:
        for k in lst:
```

```
            if not (i==j==k==0):
                mstruc.atoms = np.concatenate((mstruc.atoms, original_atoms
                + np.array([i*supercellx, j*supercelly, k*supercellz])))
                mstruc.spins = np.concatenate((mstruc.spins, original_spins))

#print('Finished boundary conditions')
mstruc.atoms = lat.cartesian(mstruc.atoms)

# create the MPDFcalculator
mc = MPDFcalculator(mstruc, rmax=6)

### Create the copydict dictionary

# dictionary giving indices of all the duplicate atoms for each atom in
    the original cell.
# This will be useful for updating the spins when a random spin flip is
    made.

copydict = {}
original_size = len(original_atoms)
for i in range(original_size):
    number_of_duplicates = 26 # need 26 duplicates for 2D (8 for 2D)
    lst = (np.arange(number_of_duplicates)+1)*original_size + i
    copydict[i] = lst
```

```
### Create the couplings dictionary, which pairs each spin with its 4
    nearest neighbors.
### Similar dictionaries can be created for 2nd NN interactions, etc.
### gather nearest neighbor indices to use for calculating the energies
NN1 = {}
NN2 = {}
NN3 = {}
NN4 = {}

for i, atom in enumerate(mstruc.atoms[:original_size]):
    distances = np.apply_along_axis(np.linalg.norm, 1, atom - mstruc.atoms)
    distances = np.round(distances, decimals=4)

pickDist = list(set(distances))
pickDist.sort()

NNdistance1 = pickDist[1] #insert half the c axis
NNdistance2 = pickDist[2] # a/c lattice parameter
NNdistance3 = pickDist[3]
NNdistance4 = pickDist[4]
print('nn1 = ', NNdistance1)
print('nn2 = ', NNdistance2)
print('nn3 = ', NNdistance3)
print('nn4 = ', NNdistance4)
for i, atom in enumerate(mstruc.atoms[:original_size]):
```

```
    distances = np.apply_along_axis(np.linalg.norm, 1, atom - mstruc.atoms)
    distances = np.round(distances, decimals=4)
    mask1 = np.abs(distances - NNdistance1) < 0.0001
    NN1[i] = np.arange(len(mstruc.atoms))[mask1]
    mask2 = np.abs(distances - NNdistance2) < 0.1
    NN2[i] = np.arange(len(mstruc.atoms))[mask2]
    mask3 = np.abs(distances - NNdistance3) < 0.1
    NN3[i] = np.arange(len(mstruc.atoms))[mask3]
    mask4 = np.abs(distances - NNdistance4) < 0.1
    NN4[i] = np.arange(len(mstruc.atoms))[mask4]

#FOR MNTE J1=21.5 K, J2=0.67 K, and J3=2.87 K # fix this J value
#multiply 2006 paper values by 2 because we put the factor of 1/2 in the
        hamiltonian
kB = .08617 #meV/K
####2006 PAPER VALUES
#j1=-21.5*2
#j2=0.67*2
#j3=-2.87*2



#####New Paper Values given in meV so devide by kB
j1=-4.135/kB
j2=-0.025/kB
j3=-0.55/kB
```

```
j4=-0.2175/kB

# create the MPDFcalculator
mc = MPDFcalculator(mstruc, rmax=15)

#4x4x4 grid energy
E0=calcEnergy(mstruc, NN1,NN2,NN3,NN4, j1, j2, j3,j4)

print(E0)

# Do lots of simulations at given temperature to get reasonable statistics
reset_spins(mstruc, copydict)


Ts = [1000,800,600,500,400,380,360,350,340,330,325,320,310,300,290,280,
        275,270,260,250,240,220,200,150,100,50,20,5]

gdict = {}
energydict = {}


for T in Ts:
    gdict[T] = []
    energydict[T] = []
```

```
num_sims = 600 #5x5 500
num_steps = 30000 # max number of steps 30000 for 5x5 grid

r, g = mc.calc()

for i in range(num_sims):
    if i % 100 == 0:
        print(i)
    reset_spins(mstruc, copydict)

    for temps in Ts:
        #print(temps)
        monte_carlo_simulation(temps, num_steps, mstruc, j1, j2, j3, j4,
        energy = calcEnergy(mstruc, NN1,NN2,NN3,NN4, j1, j2, j3, j4)
        energydict[temps].append(energy)
        gdict[temps].append(mc.calc()[1])

        #print('Energydict: \n', energydict)
        #print('Gdict: \n', gdict)



with open('MnTe4x4x4EnergiesRmax15_J4_aniso.pkl', 'wb') as fp:
    pickle.dump(energydict, fp)
    print('success')
```

```
with open('MnTe4x4x4gValsRmax15_J4_aniso.pkl', 'wb') as fp:
    pickle.dump(gdict, fp)
    print('success')
```

```
fig = plt.figure()
ax = fig.add_subplot(111)
ax.plot(r, g)
```

Code used for fitting the simulations:

```
def doFitHeis(rsim, gsim, p0, showplot=True):
    ### Make the MagStructure.
    rmax = 30

    struc = loadStructure('MnTe_hexA.cif')

    mspec = MagSpecies(struc=struc,rmaxAtoms=rmax+5,ffparamkey='Mn2')
    mspec.strucIdxs = [2, 3] # determined from previous inspection of the
                                    unit cell
    mspec.basisvecs = np.array([1, -1, 0])/np.sqrt(2) # any vector
            perpendicular to the c axis will be fine
    mspec.kvecs = np.array([0, 0, 1]) # this gives the desired AF
```

```
        structure for this choice of unit cell


# set up the MagStructure
mstruc = MagStructure(rmaxAtoms=rmax+5)
mstruc.loadSpecies(mspec)
mstruc.makeAll() ### populates the spin and atom arrays
upSpins = np.apply_along_axis(np.linalg.norm, 1,
            mstruc.spins - mstruc.spins[0]) < 0.01
downSpins = ~upSpins
# choose a small enough Uiso value to be able to resolve
    distinct peaks
mstruc.Uiso = 0.005

mc = MPDFcalculator(mstruc, paraScale=0, rmin=rsim.min(), rmax=rsim.max())

def resid(p0):
    mc.ordScale, theta, phi, xi_ab, xi_c = p0
    newvec = np.array([np.sin(theta)*np.cos(phi),
                np.sin(theta)*np.sin(phi), np.cos(theta)])
    mstruc.spins[upSpins] = 1.0*newvec
    mstruc.spins[downSpins] = -1.0*newvec
    mstruc.dampingMat = np.array([[1/xi_ab**2, 0, 0],
                        [0, 1/xi_ab**2, 0],
                        [0, 0, 1/xi_c**2]])
    r, gcalc = mc.calc(correlationMethod='full')
```

```
        return gsim - gcalc
    lb = [0, 0, -np.pi, 0.2, 0.2]
    ub = [100, np.pi, np.pi, 10 * rsim.max(), 10 * rsim.max()]
    opt = least_squares(resid, p0, bounds=[lb, ub])
    rfit, gfit = mc.calc(correlationMethod='full')
    if showplot:
        fig = plt.figure()
        ax = fig.add_subplot(111)
        ax.set_xlabel(r'r ($\mathdefault{\AA}$)')
        ax.set_ylabel(r'mPDF')
        ax.plot(rsim, gsim, label='Sim Data')
        ax.plot(rfit, gfit, label='Fit')
        plt.tight_layout()
        plt.legend()
        plt.show()
    return mc, opt


def get_wstd(vals, weights):
    numerator = []
    for i in range(0, len(weights)):
        numerator.append(weights[i]*(vals[i]-get_wmean(vals, weights))**2)
    var = sum(numerator)/(((len(weights)-1)*sum(weights))/len(weights))
    wstdev = math.sqrt(var)
    return wstdev
```

```
def get_wmean(vals, weights):
    weighted_vals = []
    vals_n_weights = [(vals[i], weights[i]) for i in range(0, len(weights))]
    for tup in vals_n_weights:
        weighted_vals.append(round(tup[0]*tup[1]/sum(weights), 3))
    answer = sum(weighted_vals)
    return answer

lmops={}
xi_abss={}
xi_css={}

for T in Ts:
    lmops[T] = []
    xi_abss[T] = []
    xi_css[T]= []

for T in Ts:
    print('Finding fits for ', T, 'K')
    gs = gdict[T][:25]
    for g in gs:
        mcfit, opt = doFitHeis(r, g, p0,showplot=False)
        p0 = opt.x
        mcfit.magstruc.corrLength = p0[-1] # set c-axis correlation length
                to overall xi for calculating lmop
```

```
        lmops[T].append(calculate_ordered_moment(mcfit, 1))
        xi_abss[T].append(p0[3])
        xi_css[T].append(p0[4])

weights = np.exp(-np.abs((E0-np.array(edict[250]))/E0))

lmopUWM={}
lmopWM={}
lmopSTD={}
lmopWSTD={}
xi_absUWM={}
xi_absWM={}
xi_absSTD={}
xi_absWSTD={}
xi_csUWM={}
xi_csWM={}
xi_csSTD={}
xi_csWSTD={}

for T in Ts:
    lmopUWM[T] = []
    lmopWM[T] = []
    lmopSTD[T] = []
    lmopWSTD[T] = []
    xi_absUWM[T] = []
```

```
    xi_absWM[T] = []
    xi_absSTD[T] = []
    xi_absWSTD[T] = []
    xi_csUWM[T]= []
    xi_csWM[T]= []
    xi_csSTD[T]= []
    xi_csWSTD[T]= []

for T in Ts:
    weights = np.exp(-np.abs((E0-np.array(edict[T][:25]))/E0))
    lmopUWM[T] = sum(lmops[T])/len(lmops[T])
    lmopWM[T] = get_wmean(lmops[T],weights)
    lmopSTD[T] =  statistics.stdev(lmops[T])
    lmopWSTD[T] =  get_wstd(lmops[T], weights)

    xi_absUWM[T] = sum(xi_abss[T])/len(xi_abss[T])
    xi_absWM[T] = get_wmean(xi_abss[T],weights)

    xi_absSTD[T] =statistics.stdev(xi_abss[T])
    xi_absWSTD[T] =get_wstd(xi_abss[T], weights)

    xi_csUWM[T] = sum(xi_css[T])/len(xi_css[T])
    xi_csWM[T] = get_wmean(xi_css[T],weights)

    xi_csSTD[T] = statistics.stdev(xi_css[T])
```

```
xi_csWSTD[T] = get_wstd(xi_css[T], weights)
```

Bibliography

[1] W. Kuhs and M. Lehmann, "The structure of ice-Ih," Water Sci. Rev 2 (1986).

[2] B. A. Frandsen and H. E. Fischer, "A new spin on material properties: Local magnetic structure in functional and quantum materials," Chemistry of Materials **36,** 9089–9106 (2024).

[3] S. J. Billinge and M. G. Kanatzidis, "Beyond crystallography: the study of disorder, nanocrystallinity and crystallographically challenged materials with pair distribution functions," Chemical communications pp. 749–760 (2004).

[4] E. Dagotto, "Complexity in strongly correlated electronic systems," Science **309,** 257–262 (2005).

[5] C. A. Young and A. L. Goodwin, "Applications of pair distribution function methods to contemporary problems in materials chemistry," Journal of Materials Chemistry **21,** 6464–6476 (2011).

[6] D. A. Keen and A. L. Goodwin, "The crystallography of correlated disorder," Nature **521,** 303–309 (2015).

[7] H. Zhu, Y. Huang, J. Ren, B. Zhang, Y. Ke, A. K.-Y. Jen, Q. Zhang, X.-L. Wang, and Q. Liu, "Bridging structural inhomogeneity to functionality: pair distribution function methods for functional materials development," Advanced Science **8,** 2003534 (2021).

[8] A. Zunger, "Bridging the gap between density functional theory and quantum materials," Nature computational science **2,** 529–532 (2022).

[9] S. A. Kimber *et al.*, "Dynamic crystallography reveals spontaneous anisotropy in cubic GeTe," Nature materials **22,** 311–315 (2023).

[10] Y. Xie *et al.*, "A percolation theory for designing corrosion-resistant alloys," Nature materials **20,** 789–793 (2021).

[11] S. Chen, T. Wang, X. Li, Y. Cheng, G. Zhang, and H. Gao, "Short-range ordering and its impact on thermodynamic property of high-entropy alloys," Acta Materialia **238,** 118201 (2022).

[12] Y. Wang, W. Zhang, L. Chen, S. Shi, and J. Liu, "Quantitative description on structure–property relationships of Li-ion battery materials for high-throughput computations," Science and Technology of advanced MaTerialS **18,** 134–146 (2017).

[13] Y. Liu *et al.*, "Short-range crystalline order-tuned conductivity in Cr2Si2Te6 van der Waals magnetic crystals," ACS nano **16,** 13134–13143 (2022).

[14] J. Reimers, J. Greedan, R. Kremer, E. Gmelin, and M. Subramanian, "Short-range magnetic ordering in the highly frustrated pyrochlore Y 2 Mn 2 O 7," Physical Review B **43,** 3387 (1991).

[15] C. Broholm, G. Aeppli, G. Espinosa, and A. Cooper, "Antiferromagnetic fluctuations and short-range order in a Kagomé lattice," Physical review letters **65,** 3173 (1990).

[16] J. A. Mydosh, "Spin glasses: redux: an updated experimental/materials survey," Reports on Progress in Physics **78,** 052501 (2015).

[17] L. Clark and A. H. Abdeldaim, "Quantum spin liquids from a materials perspective," Annual Review of Materials Research **51,** 495–519 (2021).

[18] M. G. Tucker, M. T. Dove, and D. A. Keen, "Application of the reverse Monte Carlo method to crystalline materials," Journal of applied crystallography **34,** 630–638 (2001).

[19] T. Proffen, S. Billinge, T. Egami, and D. Louca, "Structural analysis of complex materials using the atomic pair distribution function—A practical guide," Zeitschrift für Kristallographie-Crystalline Materials **218,** 132–143 (2003).

[20] P. Juhás, D. Cherba, P. Duxbury, W. Punch, and S. Billinge, "Ab initio determination of solid-state nanostructure," Nature **440,** 655–658 (2006).

[21] T. Egami and S. J. Billinge, *Underneath the Bragg peaks: structural analysis of complex materials* (Elsevier, 2003), Vol. 16.

[22] M. W. Terban and S. J. Billinge, "Structural analysis of molecular materials using the pair distribution function," Chemical Reviews **122,** 1208–1272 (2021).

[23] B. A. Frandsen, X. Yang, and S. J. L. Billinge, "Magnetic pair distribution function analysis of local magnetic correlations," Acta Crystallographica Section A **70,** 3–11 (2014).

[24] R. Skomski, in *Handbook of Magnetism and Magnetic Materials*, J. M. D. Coey and S. S. Parkin, eds., (Springer International Publishing, Cham, 2021), pp. 53–102.

[25] S. Blundell, *Magnetism in condensed matter* (OUP Oxford, 2001).

[26] R. Kubo, "The spin-wave theory of antiferromagnetics," Physical Review **87,** 568 (1952).

[27] R. S. Fishman, J. A. Fernandez-Baca, and T. Rõõm, *Spin-Wave Theory and its Applications to Neutron Scattering and THz Spectroscopy* (Morgan & Claypool Publishers, 2018).

[28] C. Broholm, R. J. Cava, S. Kivelson, D. Nocera, M. Norman, and T. Senthil, "Quantum spin liquids," Science **367,** eaay0668 (2020).

[29] L. Balents, "Spin liquids in frustrated magnets," nature **464,** 199–208 (2010).

[30] Y. Zhou, K. Kanoda, and T.-K. Ng, "Quantum spin liquid states," Reviews of Modern Physics **89,** 025003 (2017).

[31] J. A. Paddison, "Spinteract: a program to refine magnetic interactions to diffuse scattering data," Journal of Physics: Condensed Matter **35,** 495802 (2023).

[32] T. Chatterji, R. Schneider, J.-U. Hoffmann, D. Hohlwein, R. Suryanarayanan, G. Dhalenne, and A. Revcolevschi, "Diffuse magnetic scattering above T_C in quasi-two-dimensional $La_{1.2}Sr_{1.8}Mn_2O_7$," Phys. Rev. B **65,** 134440 (2002).

[33] K. Binder, "The Monte Carlo method for the study of phase transitions: A review of some recent progress," Journal of Computational Physics **59,** 1–55 (1985).

[34] L. Onsager, "Crystal statistics. I. A two-dimensional model with an order-disorder transition," Physical Review **65,** 117 (1944).

[35] B. A. Cipra, "An introduction to the Ising model," The American Mathematical Monthly **94,** 937–959 (1987).

[36] B. Shalaev, "Critical behavior of the two-dimensional Ising model with random bonds," Physics reports **237,** 129–188 (1994).

[37] B. M. McCoy and T. T. Wu, *The two-dimensional Ising model* (Harvard University Press, 1973).

[38] V. S. Dotsenko and V. S. Dotsenko, "Critical behaviour of the phase transition in the 2D Ising model with impurities," Advances in Physics **32,** 129–172 (1983).

[39] L. Gonçalves, "Uniaxial anisotropy effects in the Ising model: an exactly soluble model," Physica Scripta **32,** 248 (1985).

[40] M. E. Fisher, "Simple Ising models still thrive!: A review of some recent progress," Physica A: Statistical Mechanics and its Applications **106,** 28–47 (1981).

[41] Z. Dun *et al.*, "Neutron scattering investigation of proposed Kosterlitz-Thouless transitions in the triangular-lattice Ising antiferromagnet TmMgGaO 4," Physical Review B **103,** 064424 (2021).

[42] J.-U. Lee, S. Lee, J. H. Ryoo, S. Kang, T. Y. Kim, P. Kim, C.-H. Park, J.-G. Park, and H. Cheong, "Ising-type magnetic ordering in atomically thin FePS3," Nano letters **16,** 7433–7438 (2016).

[43] A. Hashmi, K. Nakanishi, M. U. Farooq, and T. Ono, "Ising ferromagnetism and robust half-metallicity in two-dimensional honeycomb-kagome Cr2O3 layer," npj 2D Materials and Applications **4,** 39 (2020).

[44] T. Komatsubara, M. Murakami, and E. Hirahara, "Magnetic properties of manganese telluride single crystals," Journal of the Physical Society of Japan **18,** 356–364 (1963).

[45] J. Wasscher, "Evidence of weak ferromagnetism in MnTe from galvanomagnetic measurements," Solid State Communications **3,** 169–171 (1965).

[46] C. F. Squire, "Antiferromagnetism in some manganous compounds," Physical Review **56,** 922 (1939).

[47] M. Podgorny and J. Oleszkiewicz, "Electronic structure of antiferromagnetic MnTe," Journal of Physics C: Solid State Physics **16,** 2547 (1983).

[48] G. Zanmarchi, "Optical measurements on the antiferromagnetic semiconductor MnTe," Journal of Physics and Chemistry of Solids **28,** 2123–2130 (1967).

[49] J. Wasscher and C. Haas, "Contribution of magnon-drag to the thermoelectric power of antiferromagnetic Mn Te," Physics Letters **8,** 302–304 (1964).

[50] J. Endo, H. Matsuura, and M. Ogata, "Effect of paramagnon drag on thermoelectric transport properties: Linear response theory," Phys. Rev. B **105,** 045101 (2022).

[51] M. M. H. Polash, D. Moseley, J. Zhang, R. P. Hermann, and D. Vashaee, "Understanding and design of spin-driven thermoelectrics," Cell Reports Physical Science **2,** 100614 (2021).

[52] R. Baral *et al.*, "Real-space visualization of short-range antiferromagnetic correlations in a magnetically enhanced thermoelectric," Matter **5,** 1853–1864 (2022).

[53] L. Šmejkal, J. Sinova, and T. Jungwirth, "Emerging research landscape of altermagnetism," Physical Review X **12,** 040501 (2022).

[54] D. Kriegner *et al.*, "Magnetic anisotropy in antiferromagnetic hexagonal MnTe," Physical Review B **96,** 214418 (2017).

[55] G. Yumnam *et al.*, "Magnon gap tuning in lithium-doped MnTe," Physical Review B **109,** 214434 (2024).

[56] R. Baral, Ph.D. thesis, Brigham Young University, 2022.

[57] B. A. Frandsen, P. K. Hamilton, J. A. Christensen, E. Stubben, and S. J. Billinge, "diffpy. mpdf: open-source software for magnetic pair distribution function analysis," Applied Crystallography **55,** 1377–1382 (2022).

[58] M. E. Newman and G. T. Barkema, "Monte Carlo study of the random-field Ising model," Physical Review E **53,** 393 (1996).

[59] Y. Laosiritaworn, S. Ananta, and R. Yimnirun, "Temperature effects in the magnetic properties of two-dimensional Ising square lattices: A Monte Carlo investigation," Phys. Rev. B **75,** 054417 (2007).

[60] A. Belemuk and S. Stishov, "Phase transitions in chiral magnets from Monte Carlo simulations," Physical Review B **95,** 224433 (2017).

[61] D. H. Moseley *et al.*, "Giant doping response of magnetic anisotropy in MnTe," Physical Review Materials **6,** 014404 (2022).

[62] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," Soft computing **22,** 387–408 (2018).

[63] A. Ramirez and S. Syzranov, "Short-range order and hidden energy scale in geometrically frustrated magnets," Materials Advances (2025).

[64] S. Bramwell *et al.*, "Spin correlations in Ho 2 Ti 2 O 7: a dipolar spin ice system," Physical Review Letters **87,** 047205 (2001).

[65] R. Sibille *et al.*, "Experimental signatures of emergent quantum electrodynamics in Pr2Hf2O7," Nature Physics **14,** 711–715 (2018).

[66] P. Asselin, R. F. L. Evans, J. Barker, R. W. Chantrell, R. Yanes, O. Chubykalo-Fesenko, D. Hinzke, and U. Nowak, "Constrained Monte Carlo method and calculation of the temperature dependence of magnetic anisotropy," Physical Review B—Condensed Matter and Materials Physics **82,** 054415 (2010).