

# Vacancy structure in Prussian blue analogues

Max Krummenacher

Semester project HS23  
BSc material science and engineering ETH Zürich

## Abstract

This will be my abstract.

## 1. Introduction

### 1.1. Prussian Blue Analogues

For the purposes of this report we consider Prussian blue analogues (PBA) to be compounds with composition  $M[M'(CN)_6]_x$  where M and M' are transition metals and x is the fill fraction of the hexacyanometalates. These idealized structure of these compounds is the face centered cubic structure shown in Figure 1.

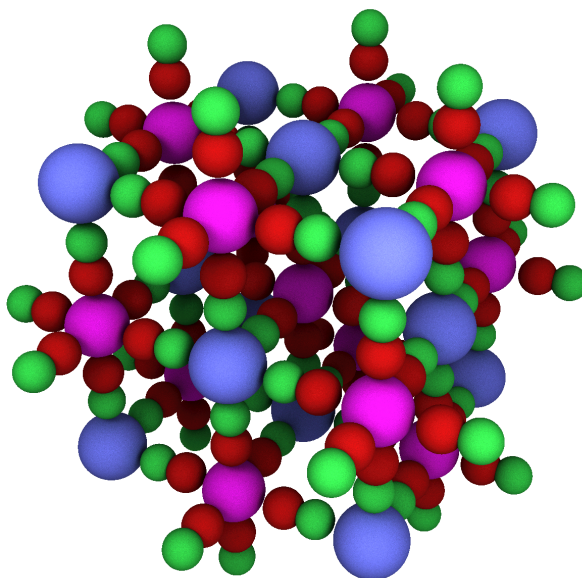


Figure 1: The structure of  $M[M'(CN)_6]$

However this exact structure is only achievable if the M ion has the same charge as the  $[M'(CN)_6]$  ion in other words if charge balance allows  $x = 1$ . If this condition is not met then some of the hexacyanometalate sites remain unoccupied. These vacancies may be filled with water or alkali metals depending on the composition and production method.

Prussian blue itself has the composition  $Fe^{III}[Fe^{II}(CN)_6]_{3/4}$ . The fill fraction of  $3/4$  allows the vacancies to be arranged in a periodic fashion in the cubic crystal structure.

In other PBAs like for example  $Mn^{II}[Co^{III}(CN)_6]_{2/3}$  the fill fraction might not allow for a periodic arrangement of vacancies in a cubic crystal. This leads the crystal to be a frustrated system, as there is no singular state with lowest energy. Some additional structure is still expected to occur as depending on the elements in the structure chains of vacancies along  $\langle 110 \rangle$  or  $\langle 100 \rangle$  might be preferred.

In this paper we explore the arrangement of vacancies of material with  $x = \frac{2}{3}$  through a Monte Carlo simulation.

## 1.2. Diffuse scattering

The Laue process is a common crystallographic method by which the Laue group of crystal and crystal parameters can be determined. In the Laue method a beam of X-rays of a band of wavelengths is diffracted by a single crystal. This causes the beam to diffract at the atoms according to the following formula:

$$F_{hkl} = \sum_j f_j \exp \left( 2\pi i (h \ k \ l) \begin{pmatrix} a_j \\ b_j \\ c_j \end{pmatrix} \right) \quad (1)$$

Where  $F_{hkl}$  is the complex structure factor at position  $(h \ k \ l)$  in the reciprocal space,  $f_j$  is the atomic form factor of the atom at position  $\begin{pmatrix} a_j \\ b_j \\ c_j \end{pmatrix}$ . In a perfect infinitely large crystal the structure factor is zero everywhere except at integer valued  $(h \ k \ l)$ . Note that the Laue method measures the intensity  $I = |F|^2$  such that the phase of  $F_{hkl}$  is lost.

The best use case for the Laue method is on a perfect single crystal. If the periodicity of the crystal is broken the peaks measured broaden and become more diffuse. However if there is some statistical pattern in the aperiodicity of the crystal (as in PBAs) the change of the shape of the peaks is characteristic of this statistical pattern.

## 1.3. Monte Carlo Simulation

For a system, of which we know all possible states  $\sigma \in \Omega$  and the hamiltonian for each state  $\mathcal{H}(\sigma)$ , we can obtain a macroscopic view of the system by the Monte Carlo method. Suppose the probability of a state occurring  $\rho(\sigma)$  at some temperature  $T > 0$  is proportional to  $\exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right)$ . We know the probabilities sum to 1.

$$1 = \sum_{\sigma \in \Omega} \frac{\exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right)}{Z} = \frac{1}{Z} \sum_{\sigma \in \Omega} \exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right) \quad (2)$$

$$\Rightarrow Z = \sum_{\sigma \in \Omega} \exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right) \quad (3)$$

thus:

$$\rho(\sigma) = \frac{\exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right)}{Z} \quad (4)$$

We define macroscopic variables of state in terms of the expected value of their associated microscopic variables. For example for the internal energy we define:

$$U = \langle \mathcal{H} \rangle = \sum_{\sigma \in \Omega} \mathcal{H}(\sigma) \rho(\sigma) \quad (5)$$

These sums often are mathematically complex and as such very hard to solve analytically. Though these sums are computable for systems with a small number of states to obtain statistically relevant results a large system needs to be chosen and the calculation of the expectance value becomes computationally

expensive. In the Monte Carlo simulation we find an approach to calculate the expected value of these variables without iterating through all states.

The fundamental idea behind the technique is to generate a series of states  $\sigma_i$  drawn from the probability distribution defined by  $\rho(\sigma)$ . Variables of state can then be calculated as the average of the microscopic variable.

For this simulation these draws are generated by the following process:

- 1) Start with some randomly generated state.
- 2) Make a small change to the state.
- 3) Compare the hamiltonians of the states.
  - a) If the hamiltonian is lower in the new state the new state is accepted.
  - b) If the hamiltonian is higher the new state is accepted with probability  $\rho = \exp\left(-\frac{\Delta\mathcal{H}}{k_b T}\right)$ .
- 4) back to step 2)

## 2. Methods

### 2.1. Simulation

The simulation is written in Rust.

#### 2.1.1. States

The structure is modeled as a three dimensional square grid containing  $N$  unit cells. To represent the fcc structure of the PBA, a grid of size  $2N \times 2N \times 2N$  is created and all positions with  $i + j + k \equiv 0(\text{mod } 2)$  are populated with 0 representing the fixed M sites. The other positions are populated with -1 or 1 representing a vacancy or a cyanometalate respectively. To achieve the exact fill fraction a vector containing the correct number of -1 and 1 is generated and then shuffled to achieve a random starting state. Note that this creates a three dimensional version of a checkerboard pattern where each direct neighbor to an -1/1 site is a 0 site. Additionally the grid is accessed in a modular fashion such that  $e_{i,j,k} = e_{i+2N,j,k} = e_{i,j+2N,k} = e_{i,j,k+2N}$ .

#### 2.1.2. Hamiltonian

The Hamiltonian of the system is divided into 2 terms. Only interactions between cyanometalates and vacancies are considered. For the nearest neighbor with an offset of  $\frac{1}{2}\langle 110 \rangle$  if both of them are the same (vacancies or metalates) their contribution to the total energy is  $J_1$  otherwise it is  $-J_1$ . Similarly for next nearest neighbors along  $\langle 100 \rangle$  directions, if both are the same their contribution is  $J_2$ ,  $-J_2$  otherwise.



Figure 2: Images showing the nearest and next nearest neighbors of M' around M left, right respectively

Let  $I = \{(i, j, k) \mid i + j + k \equiv 1(\text{mod } 2), 0 \leq i, j, k, < 2N\}$  be the set of all indexes of possible cyanometalate sites and  $\sigma_{i,j,k}$  the value at index  $(i, j, k)$ . With this notation we can express the hamiltonian in the following way:

$$\begin{aligned}\mathcal{H}(\sigma) &= \sum_{\vec{r}_0 \in I} \left( \frac{1}{2} \sum_{\vec{r} \in \langle 110 \rangle} J_1 \sigma_{\vec{r}_0} \sigma_{\vec{r}_0 + \vec{r}} + \frac{1}{2} \sum_{\vec{r} \in 2\langle 100 \rangle} J_2 \sigma_{\vec{r}_0} \sigma_{\vec{r}_0 + \vec{r}} \right) \\ &= J_1 \left( \frac{1}{2} \sum_{\vec{r}_0 \in I} \sum_{\vec{r} \in \langle 110 \rangle} \sigma_{\vec{r}_0} \sigma_{\vec{r}_0 + \vec{r}} \right) + J_2 \left( \frac{1}{2} \sum_{\vec{r}_0 \in I} \sum_{\vec{r} \in 2\langle 100 \rangle} \sigma_{\vec{r}_0} \sigma_{\vec{r}_0 + \vec{r}} \right) \\ &= J_1 s_1(\sigma) + J_2 s_2(\sigma)\end{aligned}\tag{6}$$

Where the factors of  $\frac{1}{2}$  correct for counting each neighbor interaction twice and the functions  $s_1(\sigma)$  and  $s_2(\sigma)$  correspond to the sum over nearest neighbor and next nearest neighbor respectively. (Note that here  $\langle 110 \rangle$  and  $\langle 100 \rangle$  correspond to offsets of the indexes into the array and not crystallographic vectors)

Instead of calculating the hamiltonian directly the sums  $s_1$  and  $s_2$  are computed and updated in each step. As these sums are of integer value they allow avoiding float inaccuracies.

### 2.1.3. Parameters

For each Monte Carlo step two of the vacancy/cyanometalate sites, which aren't the same are chosen uniformly from the whole grid. To calculate the difference in energy, if a swap was to be performed is calculated by calculating how the sums over nearest neighbors and next nearest neighbors. If this difference in energy is smaller than zero the swap is accepted, otherwise the swap is allowed with a probability  $p = \exp\left(-\frac{\Delta E}{k_b T}\right)$ .

As by the consideration in the appendix Equation 8 only two new parameters  $J' = \frac{J_1}{J_2}$  and  $T' = \frac{k_b T}{J_2}$  need to be explored to get an overview over the whole parameter space. For the simulation an ensemble of 20 models with different  $J'$  terms is simulated. Each model is tempered from some initial temperature  $T_0$  to  $T_{n-1}$ . The temperature steps are chosen such that  $\ln(T_i)$  is a linear function of  $i$ . At each temperature the model is left to equilibrate for 500 epochs before any measurements are taken from another 500 epochs. Where an epoch refers to  $8S$  Monte Carlo steps with the number of vacancy/cyanometalate sites  $S$ .

### 2.1.4. Output

At the end of each temperature step a `.mmCIF` file is generated from the model, which is a crystallographic file format which contains information about positions of atoms in the supercell. All values from the measurements are logged to a `.csv` file.

### 2.1.5. Optimizations

For the compiler to be able to fully optimize the program, where possible the size of list of values, like the grid is give at compile time through the use of `[i8; N]` instead of `Vec<i8>`. Though the Monte Carlo method itself is not parallelizable many systems of different  $J'$ s can be run at the same time. For this the crate `rayon` was used.

## 2.2. Data Analysis

On the `.mmCIF` files produced by the simulation the command line tool `gemmi sfcalc` is run. This program calculates the structure factors using the fast Fourier transform for the crystal provided in the input file. `gemmi` is called from a python script which then converts the data into the `Yell` format.

## 3. Results

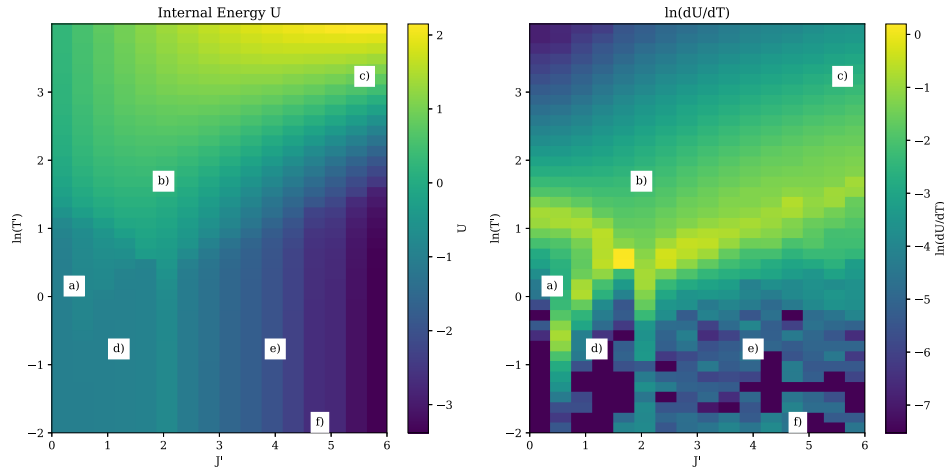


Figure 3: Internal Energy of the system and the natural log of its derivative with respect to temperature

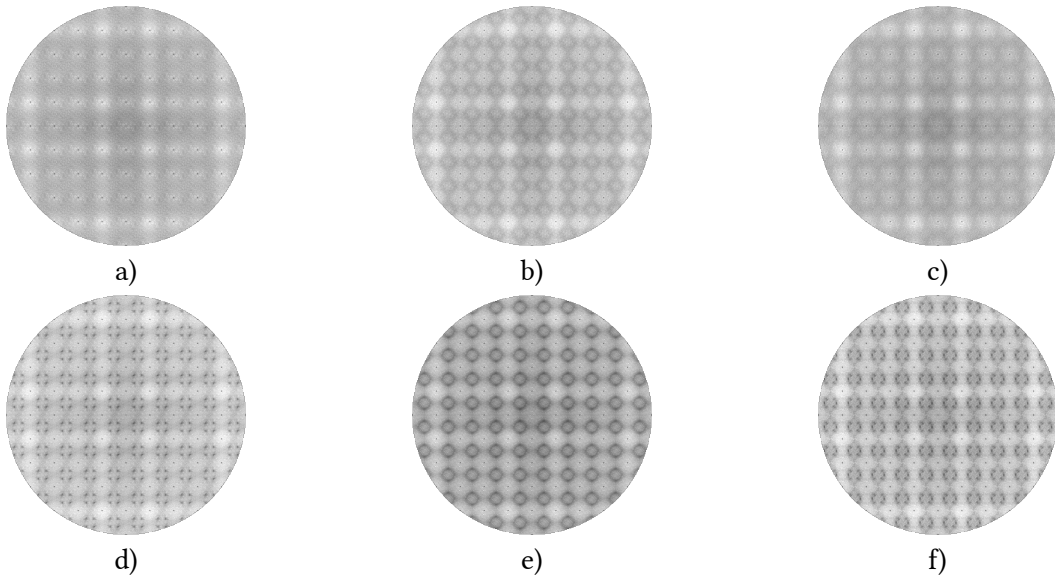


Figure 4: Diffraction patterns at the positions labeled in Figure 3

Figure 3 shows the energy calculated during the simulation. In the plot of the derivative of the internal energy invalid values resulting from the log were replaced by the minimum values.

## 4. Discussion

In Figure 3 phase transitions can be clearly recognized. The change of the structures can be seen in the diffraction pattern in Figure 4.

If further work on the simulation was to be made measurements of more different characteristic of the system might show more phase transition which cannot be interpreted from only the measurement of the energy of the system.

Furthermore the simulation could be generalized to include more distant neighbor interactions to create a better approximation of the real crystals

## 5. Appendix

### 5.1. Considerations about the parameter space

Lets consider the Boltzmann factor of a system where the hamiltonian  $\mathcal{H}$  can be described by the sum of two energies multiplied by the functions  $s_1$  and  $s_2$  on the state of the system  $\sigma$ .

$$\mathcal{H}(\sigma) = J_1 s_1(\sigma) + J_2 s_2(\sigma) \quad (7)$$

Lets define two new variables  $J'$  and  $T'$  such that  $J_2 J' = J_1$  and  $J_2 T' = k_b T$ .

$$\begin{aligned} \exp\left(-\frac{\mathcal{H}(\sigma)}{k_b T}\right) &= \exp\left(-\frac{J_1 s_1(\sigma) + J_2 s_2(\sigma)}{k_b T}\right) \\ &= \exp\left(-\frac{\cancel{J_2} J' s_1(\sigma) + \cancel{J_2} s_2(\sigma)}{\cancel{J_2} T'}\right) \\ &= \exp\left(-\frac{J' s_1(\sigma) + s_2(\sigma)}{T'}\right) \end{aligned} \quad (8)$$

Thus we can describe the whole parameter space using only  $T'$  and  $J'$ .