$J - J'$ Computational Method

Louis Sharma, Daniel Larson, Aidan Carey

September 2023

Contents

1 Lattice representations **3**
1.1 Defining the direct lattice . . . 3
1.2 Periodicity of the lattice . . . 3
1.3 Choice of the lattice . . . 3
1.4 Pairs of neighbors on the lattice . . . 4

2 Reduced basis **4**
2.1 Lattice state representation . . . 4
2.2 Generating the reduced basis . . . 4

3 Hamiltonian **5**
3.1 Hamiltonian . . . 5
3.2 Operators . . . 6
3.3 Matrix elements . . . 6

4 Translational symmetry **7**

5 Combining spin-flip symmetry and translational symmetry **9**

6 Momentum-dependent calculations **10**
6.1 Reciprocal lattice . . . 10
6.2 First Brillouin Zone and High-Syemmtry Points . . . 10

7 Calculating two-point correlators **12**
7.1 General idea . . . 12
7.2 $\hat{S}_i^z \hat{S}_j^z$ correlations . . . 12
7.3 $\hat{S}_i^x \hat{S}_j^x$ correlations . . . 12
7.4 Spin-spin correlations . . . 13
7.5 Spin structure factor . . . 13
7.6 Dimer-dimer correlations . . . 14

8 Generating matrices **15**
8.1 Storing the matrix to disk . . . 15
8.2 On-the-fly approach . . . 15

9 Block diagonalizing the hamiltonian using translation and spin-flip eigenstates **16**
9.1 Spin-flip / Parity 18

1 Lattice representations

1.1 Defining the direct lattice

We work in a finite supercell of the hexagonal lattice spanned by the vectors:

$$\begin{pmatrix}\mathbf{A}_1\\ \mathbf{A}_2\end{pmatrix} = \begin{pmatrix}m_1 & n_1\\ m_2 & n_2\end{pmatrix}\begin{pmatrix}\mathbf{a}_1\\ \mathbf{a}_2\end{pmatrix} = \mathbf{M}\mathbf{a}. \tag{1.1}$$

Where $\mathbf{a}_1$ and $\mathbf{a}_2$ are the primitive lattice vectors of the honeycomb lattice which we choose to be:

$$\mathbf{a}_1 = a\mathbf{u}_x, \quad \mathbf{a}_2 = \frac{a}{2}\mathbf{u}_x + \frac{a\sqrt{3}}{2}\mathbf{u}_y. \tag{1.2}$$

and m_1, m_2, n_1, n_2 are integers. Note that the number of lattice sites is given by $M = 2(m_1n_2 - m_2n_1)$ which is $2\det\mathbf{M}$.

Once the cell has been chosen, we need to identify the lattice sites that lie within the parallelogram spanned by $\mathbf{A}_1, \mathbf{A}_2$. Since the honeycomb lattice is bipartite, we know that the sites on the A and B sub-lattices have positions:

$$\mathbf{R}_A = k\mathbf{a}_1 + l\mathbf{a}_2, \quad \mathbf{R}_B = \mathbf{R}_A + \mathbf{b}_1, \quad k, l \in \mathbb{Z}. \tag{1.3}$$

The standard method is to pick a range of values (k, l) and expand the vector $\mathbf{R}_A$ as a linear combination of the supercell vectors: $k\mathbf{a}_1 + l\mathbf{a}_2 = \mu\mathbf{A}_1 + \lambda\mathbf{A}_2$. Any point the parallelogram can be written in such a linear combination, with $0 \leq \mu < 1$ and $0 \leq \lambda < 1$. So we check if the expansion coefficients satisfy this property. If they do, the site is a part of the direct lattice. We can write the supercell expansion coefficients:

$$\begin{pmatrix}k\mathbf{a}_1\\ l\mathbf{a}_2\end{pmatrix} = \begin{pmatrix}k & 0\\ 0 & l\end{pmatrix}\mathbf{M}^{-1}\begin{pmatrix}\mathbf{A}_1\\ \mathbf{A}_2\end{pmatrix} \Rightarrow \mu = \frac{2(kn_2 - lm_2)}{M}, \quad \lambda = \frac{2(lm_1 - kn_1)}{M}. \tag{1.4}$$

1.2 Periodicity of the lattice

The supercell defined by equation 1.1 has periodic boundary conditions, so we can imagine that we are tiling our space with infinitely many replicas of the cell. Thus, any point with coordinates (k', l') that is not in the cell has an equivalent representation in it (k, l). To find this equivalent representation, we compute the supercell expansion coefficients of (k', l') and wrap them back into the cell by applying a modulo operation:

$$\mu(k, l) = \mu(k', l') \mod 1 = a, \quad \lambda(k, l) = \lambda(k', l') \mod 1 = b. \tag{1.5}$$

Which then allows us to find the coefficients (k, l) since:

$$k\mathbf{a}_1 + l\mathbf{a}_2 = a\mathbf{A}_1 + b\mathbf{A}_2 \Rightarrow k = (am_1 + bm_2), \quad l = (an_1 + bn_2) \tag{1.6}$$

1.3 Choice of the lattice

There are numerous different types of cell that one could construct. We choose to use cells that have the nice property that every site on the A sub-lattice (and consequently the B sub-lattice) can be reached by consecutive translations by $\mathbf{R} = \mathbf{a}_1$. The number of these translations is equal to the number of unit cells supported by the supercell. As a result, finding the points of the lattice is very simple: we pick the points $(k, 0)$ with $k = 0, \ldots M/2 - 1$ and apply the periodic function defined above. This gives us all the A sub-lattice sites. For the B sublattice sites, we repeat the same procedure with $(k + 1/3, 0)$. Add more details on the nature of the cells... , more references are needed

1.4 Pairs of neighbors on the lattice

An important quantity that is needed for studying certain observable are the pairs of kth neighbors on the lattice (usually first or second). To find these pairs, we need a metric for evaluating the distance on the lattice between two points $\mathbf{R} = (k, l), \mathbf{R}' = (k', l')$. In general this is given by the standard Euclidean distance. However, since we're using periodic boundary conditions, we need to pick the distance between two points $d(\mathbf{R}, \mathbf{R}')$ such that it is the smallest amongst all replicas of the cell. This is done by finding the supercell expansion coefficients of $\mathbf{R}' - \mathbf{R}$ and counting how many times they wrap around the torus. This is done via:

$$\mu \to \mu - [\mu], \quad \lambda \to \lambda - [\lambda], \tag{1.7}$$

where $[\cdot]$ denotes the nearest integer. The effect of doing this is that it will bring the supercell expansion coefficients to the range $[-0.5, 0.5]$. This works because the maximum distance between two points in the $\mathbf{A}_1$ and $\mathbf{A}_2$ directions is $|\frac{1}{2}\mathbf{A}_1|$ and $|\frac{1}{2}\mathbf{A}_2|$ respectively. This procedure gives us the shortest vector $\mathbf{R}' - \mathbf{R}$ amongst all replicas of the cell, from which we can compute the distance.

To get the pairs of kth neighbors on the lattice, we compute all the pairwise distances, sort them and extract the kth group.

2 Reduced basis

2.1 Lattice state representation

Our basis states are represented by a vector that corresponds to the values seen along a zig-zagging horizontal traversal of the lattice.

$$(s_1, s_2, s_3, \dots s_n)$$

where n is the total number of states in our unit cell. Because of the translational symmetry of the honeycomb lattice,

$$(s_1, s_2, s_3, \dots s_n) \sim T_i(s_1, s_2, s_3, \dots s_n),$$

where T_i is the operator that corresponds to a cyclic permutation (hereafter referred to as a "rotation") of the whole vector by $2i$.

Because adjacent nodes in the vector representation are adjacent on the lattice and the lattice itself is bipartite, we can relabel of a state like so:

$$(a_1, b_1, a_2, b_2, \dots, a_{\frac{n}{2}}, b_{\frac{n}{2}}).$$

This also suggests a straightforward representation of the state as two vectors corresponding to the values on each sublattice:

$$((a_1, a_2 \dots, a_{\frac{n}{2}}), (b_1, b_2, \dots, b_{\frac{n}{2}})).$$

In this representation the translation operator T_i acts on each sublattice separately, and rotates each vector by i.

2.2 Generating the reduced basis

For our reduced basis, we want to store a single representative for every translational equivalence class, and define a set of rules that determine which member of the equivalence class will be its representative. The rule we use is as follows: the translational representative is the translation whose combined vector - the A sublattice vector concatenated with the B sublattice vector - is the smallest lexicographically. As an example, if we have the lattice state $((1, -1, 0), (0, 1, -1))$, the space of possible translations is

$$((1, -1, 0), (0, 1, -1)), ((0, 1, -1), (-1, 0, 1)), ((-1, 0, 1), (1, -1, 0)).$$

The smallest state lexicographically, and therefore this class's representative, is $((-1,0,1),(1,-1,0))$, because it is the only one with a -1 in its first position.

We can then restate the class representative rule as follows: the state which is the translational representative is the state whose A sublattice vector is smallest lexicographically, if more than one state has the same A sublattice vector, the representative state is the state within this subset of translations whose B sublattice vector is smaller lexicographically (this is a restatement of the observation that "**soc**cer" precedes "**soc**ked" alphabetically because that is the alphabetical order of the words' suffixes). This restatement of the rule makes clear an observation that is key to creating the reduced basis: a representative's A sublattice vector will always be the smallest lexicographical order cyclic permutation of that vector.

This property of being the cyclic permutation of a vector with the lowest lexicographical order is also known as being a "necklace". There exists an algorithm, found by Joe Sawada, that is capable of generating these "necklaces" directly. Therefore, for a particular lattice configuration (number of holes, up electrons, and down electrons), we can directly generate the set of every possible A sublattice vector that may exist in our set of translationally equivalent representatives by looping through the possible count of holes, up electrons, and down electrons that could compose the A sublattice and generating the necklace vectors that correspond to each combination. We will call this set of vectors $\{A_i\}$. Then, for every vector $\{A_i\}$ in this set, we must find the set of vectors $\{B_{i,j}\}$ where $(A_i, B_{i,j})$ is a group representative for all $j \in (1, 2 \ldots, |B_{i,j}|)$. The union of all these vectors will contain the complete set of translationally equivalent states.

The procedure for finding the $\{B_{i,j}\}$ is as follows:

1. if the A sublattice vector A_i is not translationally symmetric, compute the vectors corresponding to every possible permutation of holes and electrons that could exist in the B sublattice given the contents of A_i and that will be $\{B_{i,j}\}$

2. if the A sublattice vector A_i is translationally symmetric*, find the length of the symmetric unit l_s that comprises A_i (e.g. for $(1,0,-1,1,0,-1)$ it would be 3). Then, compute the vectors corresponding to every possible permutation of holes and electrons that could exist in the B sublattice given the contents of A_i. From this set of vectors, keep only the ones that are unique up to a translation by l_s by choosing the lowest lexicographical order vector from each equivalence class. This set of vectors unique up to a T_{l_s} will be $\{B_{i,j}\}$.

By taking the union of these subsets of representatives $\{A_i B\}$ corresponding to a particular A sublattice vector A_i, we yield the total set of representative vectors.

*This is necessary to account for the fact that the B sublattice's vector is the lexicographic tiebreaker in determining the representative state. If some arbitrary state $((A,B))$ can be translated to (A', B'), (A', B''), (A', B''') where A' is a necklace vector, then the representative of this state will be the necklace A' composed with whichever of the B sublattice vectors (B', B'', B''') is smallest lexicographically.

3 Hamiltonian

3.1 Hamiltonian

The $J - J'$ antiferromagnet Hamiltonian is given by:

$$\hat{\mathcal{H}} = J \sum_{\langle i,j \rangle} \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j + J' \sum_{\langle\langle i,j \rangle\rangle} \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j. \tag{3.1}$$

The scalars J and J' are the first and second neighbor antiferromagnetic exchange couplings, $\langle i,j \rangle$ and $\langle\langle i,j \rangle\rangle$ are signify that the sums runs over first and second neighbors pairs of site respectively. Finally, $\hat{\mathbf{S}}_i$ is the spin operator

for site i. We can pull out a factor of $1/4$ from the above Hamiltonian to represent it in terms of Pauli operators:

$$\hat{\mathcal{H}} = J \sum_{\langle i,j \rangle} \hat{\boldsymbol{\sigma}}_i \cdot \hat{\boldsymbol{\sigma}}_j + J' \sum_{\langle\langle i,j \rangle\rangle} \hat{\boldsymbol{\sigma}}_i \cdot \hat{\boldsymbol{\sigma}}_j. \tag{3.2}$$

$\hat{\boldsymbol{\sigma}}_i = (\hat{\sigma}_i^x, \hat{\sigma}_i^y, \hat{\sigma}_i^z)$ is the vector of Pauli matrices for the spin at site i that satisfy the usual SU(2) algebra: $[\hat{\sigma}_j^\alpha, \hat{\sigma}_k^\beta] = 2i\epsilon_{\alpha\beta\gamma}\hat{\sigma}_j^\gamma \delta_{jk}$

3.2 Operators

To construct the operators, we work in the binary representation of the Fock basis, where each state $|n\rangle$ can be encoded as a binary number:

$$n = \sum_j 2^j \sigma_j \tag{3.3}$$

where $\sigma_j \in \{0, 1\} = \{\downarrow, \uparrow\}$. The numbers σ_j can be found using:

$$\sigma_j = \frac{n}{2^j} \mod 2. \tag{3.4}$$

This encoding of the states in the basis as integers is more memory efficient (compared to storing arrays of 0s and 1s) and allows us to apply the operators in the Hamiltonian using numerical operations rather than array modification. For example: $\hat{\sigma}_i^x \hat{\sigma}_j^x$ transforms the state encoded as n into n' such that:

$$n' = n - 2^i\sigma_i - 2^j\sigma_j + 2^i\sigma_j + 2^j\sigma_i = n + 2^i(\sigma_j - \sigma_i) + 2^j(\sigma_i - \sigma_j). \tag{3.5}$$

The Hamiltonian conserves the z component of spin, so the operation above is applied if and only if $\sigma_i + \sigma_j = 1$ (modulo 2 addition):

$$\hat{\sigma}_x^i \hat{\sigma}_x^j |n\rangle = \begin{cases} |n + (2\sigma_i - 1)2^i + (2\sigma_j - 1)2^j\rangle & \text{if } \sigma_i + \sigma_j = 1 \\ 0 \text{ otherwise.} \end{cases} \tag{3.6}$$

For the $\hat{\sigma}_y^i \hat{\sigma}_y^j$, we notice that the phase picked up will always be one since the z component of the spin is conserved, so we lazily write that $\hat{\sigma}_y^i \hat{\sigma}_y^j = \hat{\sigma}_x^i \hat{\sigma}_x^j$. Moreover, $\hat{\sigma}_z^i \hat{\sigma}_z^j$ leaves $|n\rangle$ unchanged but introduces a phase factor, the such that the new state is:

$$\hat{\sigma}_z^i \hat{\sigma}_z^j |n\rangle = (2\sigma_i - 1)(2\sigma_j - 1) |n\rangle . \tag{3.7}$$

3.3 Matrix elements

Now that we have a way to implement the various operators appearing the Hamiltonian, we use the following algorithm to construct its matrix representation. We construct two matrices: one for the first neighbour interactions and one for the second neighbour interactions. The process for both matrices is identical:

1. Loop through each state of the basis $|n\rangle$.
2. Loop through the pairs of kth (1 or 2) neighbours (i, j).
3. For each state and for each pair of neighours add the quantity from equation 3.7 to the diagonal.

4. For each state and each pair, check if $\sigma_i + \sigma_j = 1$. If this is the case, compute the new state labelled n' using 3.5. Then, hit the hashmap to find the index of n' in the basis and set $\langle n'|\mathcal{H}|n\rangle = 2$. If not, do nothing and continue to the next pair of neighbours.

We then store the row, column and values and save the matrices as sparse matrices, which can be used for diagonalization with the Lanczos algorithm.

4 Translational symmetry

The $J - J'$ Hamiltonian 3.2 commutes with the translation operator $\hat{T}(\mathbf{R})$ that shifts states by a lattice vector $\mathbf{R}$. This means that the eigenstates of $\hat{\mathcal{H}}$, which we denote $|\psi\rangle$ are simultaneous eigenvectors of $\hat{T}$. Since $\hat{T}^{M/2} = 1$, the eigenvalues of the translation operator can be written as roots of unity $e^{i\mathbf{k}\cdot\mathbf{R}}$. Hence, the action of the translation operator on an energy eigenstate is:

$$\hat{T}\,|\psi\rangle = e^{i\mathbf{k}\cdot\mathbf{R}}\,|\psi\rangle\,.$$

Consider the expansion of $|\psi\rangle$ in the Fock basis, with coefficients $\langle n|\psi\rangle$. If we know one such coefficent for a state $|n\rangle$, then we have:

$$\langle n|\hat{T}^\dagger|\psi\rangle = e^{-i\mathbf{k}\cdot\mathbf{R}}\,\langle n|\psi\rangle\,,$$

which means that knowing one of the expansion coefficient for a state allows us to determine the expansion coefficient for all the states that are translations of $|n\rangle$. This motivates us to work in a reduced basis, spanned by the different eigenstates of $\hat{T}$. Nice explanation from Daniel, which I copy here in my own words. In general, we assume the crystal is macroscopic, that is that it contains infinitely many unit cells. Hence, the allowed $\mathbf{k}$ values form a continuum. In our case, we're working with a finite lattice and have the condition that:

$$\hat{T}^{M/2}\,|\psi\rangle = |\psi\rangle\,, \tag{4.1}$$

which puts the constraint the $e^{iM\mathbf{k}\cdot\mathbf{R}/2} = 1$. This means that $\mathbf{k}$ must be a reciprocal lattice vector that is supported by the direct lattice supercell.

Given a state $|n\rangle$, we find the states that can be reached from it by applying powers of $\hat{T}$, which we call a *cycle*. In our case, we define the action of $\hat{T}$ with zero momentum as shifting the spins along the $\mathbf{a}_1$ direction, which corresponds to moving the elements of a Fock state by 2 places to the right. As an example, the states:

$$|\uparrow\downarrow\uparrow\downarrow\uparrow\rangle\,, |\downarrow\uparrow\uparrow\downarrow\downarrow\uparrow\rangle\,, |\downarrow\uparrow\downarrow\uparrow\uparrow\downarrow\rangle$$

form a cycle in the 6 site system. With our numerical scheme, the action of the operator is given by shifting the bits of the of integer encoding of $|n\rangle$. Given a cycle, we just need to store one state which we call the group representative and denote with it a bar: $|\bar{n}\rangle$. The group representative is chosen to be the element of the cycle that has the smallest integer representation. From this, we can construct the properly symmetrized eigenstates of the translation operator for a given value of $\mathbf{k}$:

$$|\bar{n}(\mathbf{k})\rangle = \frac{1}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}}\hat{T}^l\,|\bar{n}\rangle\,. \tag{4.2}$$

These are made by summing the states of the cycle with a phase prefactor. These are indeed eigenstates of the translation operator since:

$$\hat{T}\,|\bar{n}(\mathbf{k})\rangle = \frac{1}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}} \hat{T}^{l+1}\,|\bar{n}\rangle = \frac{e^{i\mathbf{k}\cdot\mathbf{R}}}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2-1} e^{-i(l+1)\mathbf{k}\cdot\mathbf{R}} \hat{T}^{l+1}\,|\bar{n}\rangle = e^{i\mathbf{k}\cdot\mathbf{R}}\,|\bar{n}(\mathbf{k})\rangle\,,$$

where we used the periodicity of the lattice to change the summation index $l+1 \to l$. Particular care must be given to the normalization factor appearing in equation 4.2. The sum contains $M/2$ terms but not all cycles have the same periodicity than that of the supercell (for example, the fully antiferromagnetic state always translates to itself). The periodicity N_n is defined such that $\hat{T}^{N_n}\,|\bar{n}\rangle = |\bar{n}\rangle$. If $N_n < M/2$, then certain states will appear several times in 4.2. Moreover, not all cycles of the basis will contribute to an eigenstate $|\psi\rangle$ for a given value of $\mathbf{k}$. The only ones that contribute are those that verify the commensurability condition: $e^{iN_n\mathbf{k}\cdot\mathbf{R}} = 1$, i.e $N_n\mathbf{k}\cdot\mathbf{R}$ is an integer multiple of 2π. The states that don't satisfy this condition should be removed from the new basis. The normalization factor can thus be written as:

$$\mathcal{N}_n = \frac{M^2}{4N_n}, \tag{4.3}$$

to ensure that $\langle\bar{n}(\mathbf{k})|\bar{n}(\mathbf{k})\rangle = 1$. Note that when $N_n = M/2$, we get the expected result that $\mathcal{N}_n = M/2$. The next step is to construct the Hamiltonian matrix in this basis. The matrix elements are given by $\langle\bar{m}(\mathbf{k})|\hat{\mathcal{H}}|\bar{n}(\mathbf{k})\rangle$. To implement this numerically, let's first consider the action of the Hamiltonian on one of the basis states:

$$\hat{\mathcal{H}}\,|\bar{n}(\mathbf{k})\rangle = \frac{1}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}} \hat{\mathcal{H}}\hat{T}^{l}\,|\bar{n}\rangle = \frac{1}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}} \hat{T}^{l}\hat{\mathcal{H}}\,|\bar{n}\rangle\,, \tag{4.4}$$

where we exploited translational symmetry in going to the last equality. Next, it is convenient to decompose $\hat{\mathcal{H}}$ into a sum of of operators $\hat{\mathcal{H}}_j$ that map single Fock states to other single Fock states (e.g, $\hat{\sigma}_i^x\hat{\sigma}_j^x$) such that $\hat{\mathcal{H}}_j\,|\bar{n}\rangle = \alpha_j\,|m_j\rangle$. In fact, this decomposition is quite intuitive since we treat the Hamiltonian term by term when doing the computation. This gives:

$$\hat{\mathcal{H}}\,|\bar{n}(\mathbf{k})\rangle = \frac{1}{\sqrt{N_n}} \sum_j \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}} \hat{T}^{l}\alpha_j\,|m_j\rangle\,. \tag{4.5}$$

The notation used is quite general: the $|m_j\rangle$ may to be a representatives and they may not even belong to a valid cycle for a given value of $\mathbf{k}$. If this is the case, we can discard the contribution from this state since it is not in the basis. If it is a valid state, then we can always write it as some translation of its representative: $|m_j\rangle = \hat{T}^{d_j}\,|\bar{m}_j\rangle$, where d_j is the number of shifts by $\mathbf{R}$ from $|\bar{m}_j\rangle$ needed to reach $|\bar{m}_j\rangle$. We now have:

$$\hat{\mathcal{H}}\,|\bar{n}(\mathbf{k})\rangle = \frac{1}{\sqrt{N_n}} \sum_j \sum_{l=0}^{M/2-1} e^{-il\mathbf{k}\cdot\mathbf{R}} \hat{T}^{l+d_j}\alpha_j\,|\bar{m}_j\rangle = \sum_j \frac{\alpha_j e^{id_j\mathbf{k}\cdot\mathbf{R}}}{\sqrt{\mathcal{N}_n}} \sum_{l=0}^{M/2} e^{-i(l+d_j)\mathbf{k}\cdot\mathbf{R}} T^{l+d_j}\,|\bar{m}_j\rangle\,. \tag{4.6}$$

Again, we can use the periodicity of the lattice to change the summation index $l + d_j \to l$ in the last term. We can recognize this as the state $|\bar{m}_j(\mathbf{k})\rangle$ up to a numerical factor of $\sqrt{\mathcal{N}_m}$. This yields the final result that:

$$\hat{\mathcal{H}}\,|\bar{n}(\mathbf{k})\rangle = \sum_j \alpha_j e^{id_j\mathbf{k}\cdot\mathbf{R}} \sqrt{\frac{\mathcal{N}_m}{\mathcal{N}_n}}\,|\bar{m}_j(\mathbf{k})\rangle \boxed{\Rightarrow\ \langle\bar{m}_j(\mathbf{k})|\hat{\mathcal{H}}_j|\bar{n}(\mathbf{k})\rangle = \langle m_j|\hat{\mathcal{H}}_j|\bar{n}\rangle\, e^{id_j\mathbf{k}\cdot\mathbf{R}} \sqrt{\frac{N_n}{N_m}}.} \tag{4.7}$$

For a diagonal operator, the matrix element will simply be $\langle\bar{n}|\hat{\mathcal{H}}|\bar{n}\rangle$ since $d_j = 0$.

5 Combining spin-flip symmetry and translational symmetry

When working in the $S^z = 0$ symmetry sector, Hamiltonian also commutes with the operator $\hat{F}$ that flips spins. This operator has eigenvalues $\sigma = \pm 1$. This means that for any eigenstate $|\psi\rangle$, $\hat{F}|\psi\rangle = \sigma|\psi\rangle$. The eigenstates of $\hat{F}$ can be written as:

$$|\bar{n}(\sigma)\rangle = \frac{1}{\sqrt{2}}\left(|\bar{n}\rangle + \sigma\hat{F}|\bar{n}\rangle\right). \tag{5.1}$$

Moreover, $\hat{F}$ and $\hat{T}$ also commute. This means that the eigenstates $|\psi\rangle$ can be simultaneously characterized by the quantum numbers $\mathbf{k}, \sigma$. We can not construct the simultaneous eigenstates of the translation and the spin-flip operator:

$$|\bar{n}(\mathbf{k},\sigma)\rangle = \frac{1}{\sqrt{\mathcal{N}_n}}\sum_{l=0}^{N_c-1} e^{-il\mathbf{k}\cdot\mathbf{R}}\hat{T}^l|\bar{n}(\sigma)\rangle. \tag{5.2}$$

Where $\bar{n}$ is the group representative, defined as $\min\{\hat{T}^l|n\rangle, \hat{F}\hat{T}^l|n\rangle\}_{l=0}^{N_c-1}$. We can evaluate the matrix elements of the Hamiltonian in this basis: $\langle\bar{m}(\mathbf{k},\sigma)|\hat{\mathcal{H}}|\bar{n}(\mathbf{k},\sigma)\rangle$ by evaluating the action of the Hamiltonian on 4.2. This gives:

$$\begin{aligned}\hat{\mathcal{H}}|\bar{n}(\mathbf{k},\sigma)\rangle &= \frac{1}{\sqrt{\mathcal{N}_n}}\sum_j\sum_{l=0}^{N_c-1} e^{-il\mathbf{k}\cdot\mathbf{R}}\hat{T}^l\frac{1}{\sqrt{2}}\left(\hat{\mathcal{H}}_j|\bar{n}\rangle + \sigma\hat{F}\hat{\mathcal{H}}_j|\bar{n}\rangle\right)\\ &= \sum_j\frac{\alpha_j}{\sqrt{\mathcal{N}_n}}\sum_{l=0}^{N_c-1} e^{-il\mathbf{k}\cdot\mathbf{R}}\hat{T}^l|m_j(\sigma)\rangle = \sum_j\alpha_j e^{id_j\mathbf{k}\cdot\mathbf{R}}\sqrt{\frac{N_n}{N_m}}|\bar{m}_j(\mathbf{k},\sigma)\rangle\end{aligned} \tag{5.3}$$

6 Momentum-dependent calculations

6.1 Reciprocal lattice

Since we're working with a finite cell with periodic boundary conditions (no continuum limit), there are restrictions on the values of $\mathbf{k}$, namely, they need to be points on the reciprocal lattice. The reciprocal lattice is for our finite system is defined by the supercell reciprocal vectors that are linear combinations of the reciprocal lattice vectors:

$$\begin{pmatrix}\mathbf{B}_1\\ \mathbf{B}_2\end{pmatrix} = \begin{pmatrix}m_1' & n_1'\\ m_2' & n_2'\end{pmatrix}\begin{pmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{pmatrix} \tag{6.1}$$

with:

$$\mathbf{b}_1 = \frac{2\pi}{a}\left(\mathbf{u}_x - \frac{1}{\sqrt{3}}\mathbf{u}_y\right), \quad \mathbf{b}_2 = \frac{4\pi}{\sqrt{3}a}\mathbf{u}_y. \tag{6.2}$$

However, the condition $\mathbf{A}_i \cdot \mathbf{B}_j = 2\pi\delta_{ij}$ gives four equations which fix the coefficients of the linear transformation in 6.1:

$$\begin{cases} m_1'm_1 + n_1'n_1 = 1 \\ m_1'm_2 + n_1'n_2 = 0 \end{cases}, \quad \begin{cases} m_2'm_1 + n_2'n_1 = 0. \\ m_2'm_2 + n_2'n_2 = 1. \end{cases} \tag{6.3}$$

This gives the following reciprocal lattice supervectors:

$$\begin{pmatrix}\mathbf{B}_1\\ \mathbf{B}_2\end{pmatrix} = \frac{2}{M}\begin{pmatrix}n_2 & -m_2\\ -n_1 & m_1\end{pmatrix}\begin{pmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{pmatrix}. \tag{6.4}$$

6.2 First Brillouin Zone and High-Syemmtry Points

We're interested in finding cells that contain allowed $\mathbf{k}$ points along the high symmetry points on the boundaries of the first Brillouin Zone (FBZ), such as $\mathbf{K}, \mathbf{K}'$ and $\mathbf{M}, \mathbf{M}'$[1] With our chosen convention, the coordinates of the point K can be written as:

$$\mathbf{K} = \frac{2}{3}\mathbf{b}_1 + \frac{1}{3}\mathbf{b}_2. \tag{6.5}$$

We thus want to find the conditions on the coefficients of the supercell which give:

$$\begin{aligned} p\mathbf{B}_1 + q\mathbf{B}_2 &= \frac{2}{3}\mathbf{b}_1 + \frac{1}{3}\mathbf{b}_2 \\ (pn_2 - qn_1)\mathbf{b}_1 + (qm_1 - pm_2)\mathbf{b}_2 &= \frac{M}{3}\mathbf{b}_2 + \frac{M}{6}\mathbf{b}_2, \end{aligned} \tag{6.6}$$

where p and q are integers. We immediately see that since M is always even (the lattice is bipartite) the above equation implies that M has to be divisible by 3. We also get two new equations:

$$pn_2 - qn_1 = \frac{M}{3}, \quad qm_1 - pm_2 = \frac{M}{6}. \tag{6.7}$$

The coordinates of the point $\mathbf{M}$ can be expressed in terms of the reciprocal lattice vectors using:

$$\mathbf{M} = \frac{\mathbf{b}_1}{2} \tag{6.8}$$

[1] I'm using a bold $\mathbf{M}$ to not confuse it with the number of sites, which we labelled M.

which gives the condition

$$pn_2 - qn_1 = M, \quad qm_1 - pm_2 = 0 \tag{6.9}$$

7 Calculating two-point correlators

7.1 General idea

Suppose we have a two body operator acting on a pair of sites: $\hat{A}_i\hat{B}_j$ such that its action on a basis state only returns a another basis element with a prefactor:

$$\hat{A}_i\hat{B}_j\,|n\rangle = \alpha_{nn'}\,|n'\rangle\,. \tag{7.1}$$

We are interested in computing the expectation value:

$$\begin{aligned}\langle\psi|\hat{A}_i\hat{B}_j|\psi\rangle &= \sum_{n,m}\langle\psi|m\rangle\,\langle m|\hat{A}_i\hat{B}_j|n\rangle\,\langle n|\psi\rangle \\ &= \sum_{n,m} c_m^* c_n \alpha_{nn'}\,\langle m|n'\rangle \\ &= \sum_{n} c_{n'}^* c_n \alpha_{nn'}.\end{aligned} \tag{7.2}$$

Given a state $|\psi\rangle$ we can calculate the correlation function by looping through the basis, acting $\hat{A}_i\hat{B}_j$ on each state, finding each $|n'\rangle$ and each $\alpha_{nn'}$, this avoids having to store a matrix for first, second neighbors... Repeating this procedure for every site i, j we get a correlation matrix:

$$C_{ij} = \langle\hat{A}_i\hat{B}_j\rangle\,. \tag{7.3}$$

7.2 $\hat{S}_i^z\hat{S}_j^z$ correlations

This is an example of a diagonal correlation function. The correlation is: To calculate it, we expand the dot product in the Fock basis:

$$\langle\hat{S}_i^z\hat{S}_j^z\rangle = \sum_{n,m} c_n^* c_m\,\langle n|\hat{S}_i^z\hat{S}_j^z|m\rangle = \frac{1}{4}\sum_n |c_n|^2\sigma_i^{(n)}\sigma_j^{(n)}. \tag{7.4}$$

It is just the sum of the modulus squared of the expansion coefficients weighted by the eigenvalue of the operator for the corresponding Fock state. To get the full correlation matrix, we notice first that we only need to calculate the expectation values for the upper triangle $i < j$. This is because $\hat{S}_i^z\hat{S}_j^z = \hat{S}_j^z\hat{S}_i^z$ and the diagonals $\langle\hat{S}_i^{z2}\rangle = 1/4$. We can also reduce the number of dot products that need to be calculated since:

$$\langle\hat{S}_{i+R}^z\hat{S}_{j+R}^z\rangle = \langle\hat{T}\hat{S}_i^z\hat{S}_j^z\hat{T}^\dagger\rangle = \langle\hat{S}_i^z\hat{S}_j^z\rangle\,, \tag{7.5}$$

since we take the expectation value under and eigenstate of the Hamiltonian, which is also an eigenstate of the translation operator. To get the full correlation matrix, we only need to calculate $\langle\hat{S}_0^z\hat{S}_r^z\rangle$ and $\langle\hat{S}_1^z\hat{S}_r^z\rangle$ which are the correlation functions for one site on the A sublattice and one site on the B sublattice.

7.3 $\hat{S}_i^x\hat{S}_j^x$ correlations

To calculate this, we use equation 7.2 and observe that all nonzero $\alpha_{nn'}$ are equal to 1. We can exploit the symmetries like in the previous section.

7.4 Spin-spin correlations

The spin-spin correlation function can be calculate from the $x-x$ and $z-z$ correlations using:

$$\langle \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j \rangle = \langle \hat{S}_i^z \hat{S}_j^z + 2\hat{S}_i^x \hat{S}_j^x \rangle \,. \tag{7.6}$$

This is because $\hat{S}_i^x \hat{S}_j^x + \hat{S}_i^y \hat{S}_j^y$ only returns a non-null state if sites i and j are of opposite spin, in which case the phase from applying $\hat{S}_i^y \hat{S}_j^y$ will be 1.

7.5 Spin structure factor

The spin structure factor can be defined as the Fourier transform of the spin-spin correlation function:

$$S_{\mu\nu}(\mathbf{k}) = \frac{1}{N_c} \sum_{i \in \mu, j \in \nu} e^{-i\mathbf{k}\cdot(\mathbf{R}_i - \mathbf{R}_j)} \langle \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j \rangle \,, \tag{7.7}$$

where $\mathbf{k}$ is a momentum vector supported by the supercell, N_c is the number of primitive unit cells, $\mathbf{R}_i$ is the lattice vector from the origin to the primitive cell where particle i is located and μ, ν are indices that correspond to different sublattices (AA, AB, BA, BB).

The spin structure factor can tell us whether the phase is antiferromagnetic (AF). To show this, consider the fully AF state:

$$|AF\rangle = |\uparrow\downarrow\uparrow\downarrow \ldots\rangle \,. \tag{7.8}$$

When we calculate the expectation value of the spin-spin correlations measured in this state, we see that $\langle AF|\hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j|AF\rangle = 1/4$ for site i on the A sublattice an site j also on the A sublattice, whereas $\langle \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j \rangle = -1/4$ for site i on the A sublattice and site j on the B sublattice. Hence, the spin structure factor

$$S^{\mu\nu}(\mathbf{k}) = \pm\frac{1}{4} \sum_{i \in \mu, j \in \nu} e^{-i\mathbf{k}\cdot(\mathbf{R}_i - \mathbf{R}_j)} \tag{7.9}$$

is maximal for $\mathbf{k} = \mathbf{0}$. For an antiferromagnetic eigenstate, we should observe peaks at $\mathbf{k} = \mathbf{0}$ in the spin structure factor S^{AA} and S^{AB} with phase $+1$ for S^{AA} and phase -1 for S^{BB}. Consider a *zigzag* state of the form:

$$|ZZ\rangle = |\uparrow\uparrow\downarrow\downarrow \ldots\rangle \tag{7.10}$$

In this case, correlations between AA and AB sites would form an AF pattern.

$$S^{\mu\nu}(\mathbf{k}) = -\frac{1}{4} \sum_{i \in \mu, j \in \nu} e^{-i\mathbf{k}\cdot(\mathbf{R}_i - \mathbf{R}_j)} (-1)^{i-j}. \tag{7.11}$$

This will be maximal for $\mathbf{k} \cdot (\mathbf{R}_i - \mathbf{R}_j) = \pi(j-i)$. Given our ordering, $\mathbf{R}_i - \mathbf{R}_j$ can be written as an integer multiple of $\mathbf{a}_1$, so $\mathbf{k} = \pm\mathbf{b}_1/2$ satisfies this condition. As it turns out, $\pm\mathbf{b}_1/2$ perfectly corresponds with the high symmetry points $\mathbf{M}, \mathbf{M}'$ on the first Brillouin Zone. Hence, a zigzag ordering can be identified by peaks at the $\mathbf{M}, \mathbf{M}'$ points in the spin structure factor.

$x_{ij}x_{kl}$	$x_{ij}y_{kl}$	$x_{ij}z_{kl}$
$y_{ij}x_{kl}$	$y_{ij}y_{kl}$	$y_{ij}z_{kl}$
$z_{ij}x_{kl}$	$z_{ij}y_{kl}$	$z_{ij}z_{kl}$

Table 1: Table showing the different combinations of the components of the spins on sites labelled i, j, k, l that appear in the dimer-dimer correlation function.

7.6 Dimer-dimer correlations

The dimer-dimer correlation function is given by:

$$\langle(\hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j)(\hat{\mathbf{S}}_k \cdot \hat{\mathbf{S}}_l)\rangle - \langle\hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j\rangle \langle\hat{\mathbf{S}}_k \cdot \hat{\mathbf{S}}_l\rangle . \tag{7.12}$$

It contains a four-point correlation function which has a total of nine terms. Let's see how using symmetry arguments can help us evaluate it efficiently. We immeadiately see that the diagonal entries of the table are all equivalent to $z_{ij}z_{kl}$. Since we're working in a fixed S^z symmetry sector, $x_{ij}y_{kl} = y_{ij}x_{kl} = z_{ij}z_{kl}$. We also have $z_{ij}y_{kl} = z_{ij}x_{kl}$. This leaves us with $z_{ij}z_{kl}$, $z_{ij}x_{kl}$ and $x_{ij}z_{kl}$. If we apply a rotation in spin space about the y axis of angle $\pi/2$, z_{ij} transforms to x_{ij} and x_{ij} transforms to $(-1)(-1)z_{ij} = z_{ij}$. which means that we only need to evaluate two terms: $z_{ij}z_{kl}$ and $x_{ij}z_{kl}$. We have:

$$\langle(\hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j)(\hat{\mathbf{S}}_k \cdot \hat{\mathbf{S}}_l)\rangle = 3\langle\hat{S}_i^z\hat{S}_j^z\hat{S}_k^z\hat{S}_l^z + 2\hat{S}_i^x\hat{S}_j^x\hat{S}_k^z\hat{S}_l^z\rangle .$$

Let's look at how to evaluate these *on-the-fly*:

$$\begin{aligned}\langle\hat{S}_i^z\hat{S}_j^z\hat{S}_k^z\hat{S}_l^z\rangle &= \sum_{\boldsymbol{\sigma},\boldsymbol{\sigma}'}\psi^*(\boldsymbol{\sigma}')\psi(\boldsymbol{\sigma})\lambda_{ijkl}^{\boldsymbol{\sigma}}\delta_{\boldsymbol{\sigma}'\boldsymbol{\sigma}} \\ &= \sum_{\boldsymbol{\sigma}}|\psi(\boldsymbol{\sigma})|^2\lambda_{ijkl}^{\boldsymbol{\sigma}},\end{aligned} \tag{7.13}$$

where $\lambda_{ijkl}^{\boldsymbol{\sigma}}$ is the eigenvalue of the 4-body operator acting on a basis element. The next term is given by:

$$\begin{aligned}\langle\hat{S}_i^x\hat{S}_j^x\hat{S}_k^z\hat{S}_l^z\rangle &= \sum_{\boldsymbol{\sigma},\boldsymbol{\sigma}'}\psi^*(\boldsymbol{\sigma}')\psi(\boldsymbol{\sigma})\lambda_{ij}^{\boldsymbol{\sigma}}\langle\boldsymbol{\sigma}'|\hat{S}_i^x\hat{S}_j^x|\boldsymbol{\sigma}\rangle \\ &= \frac{1}{4}\sum_{\boldsymbol{\sigma},\boldsymbol{\sigma}'}\psi^*(\boldsymbol{\sigma}')\psi(\boldsymbol{\sigma})\lambda_{ij}^{\boldsymbol{\sigma}}\delta_{\boldsymbol{\sigma}''\boldsymbol{\sigma}'} \\ &= \frac{1}{4}\sum_{\boldsymbol{\sigma}}\psi^*(\boldsymbol{\sigma}'')\psi(\boldsymbol{\sigma})\lambda_{ij}^{\boldsymbol{\sigma}}\end{aligned} \tag{7.14}$$

8 Generating matrices

The Lanczos procedure utilizes repeated matrix vector products: $\hat{\mathcal{H}}\,|\psi\rangle = |\psi'\rangle$. We thus need to implement a function that multiplies a vector by the Hamiltonian. There are two ways of doing this. The first one consists in storing the full matrix in sparse format to disk, the other consists in generating the matrix vector products *on-the-fly.*

8.1 Storing the matrix to disk

To compute all the matrix elements, we can first expand the Hamiltonian in the Fock basis:

$$\hat{\mathcal{H}} = \sum_{\sigma,\sigma'} |\sigma\rangle \ \langle\sigma|\hat{\mathcal{H}}|\sigma'\rangle \, \langle\sigma'| \, . \tag{8.1}$$

Such that one row of the matrix is given by:

$$\langle\sigma|\,\hat{\mathcal{H}} = \sum_{\sigma'} \mathcal{H}_{\sigma\sigma'} \, \langle\sigma'| = \sum_{\sigma'} \mathcal{H}^*_{\sigma'\sigma} \, \langle\sigma'| \, . \tag{8.2}$$

Then we decompose the Hamiltonian into a sum of operators that map individual basis states to other basis states or to 0 :

$$\hat{\mathcal{H}} = \sum_j \hat{\mathcal{H}}^{(j)}. \tag{8.3}$$

Thus the full Hamiltonian construction can be summarized in the following loop:

Algorithm 1 Storing the Hamiltonian to disk

```
rows, cols, vals = empty lists
for |σ⟩ in basis do
    for all j do
        H^(j)_{σ'σ}, |σ'⟩ = Ĥ^(j) |σ⟩
        rows ← σ, cols ← σ', vals ← H^(j)*_{σ'σ}.
    end for
end for
```

8.2 On-the-fly approach

In this approach, we don't save any matrix and instead generate the transformation as we're multiplying a vector by the Hamiltonian. This can save a lot of memory but comes at the cost of slowing the code down. To see how this works, we let $|\psi'\rangle = \hat{\mathcal{H}}\,|\psi\rangle$ and take the scalar product of 8.2 with $|\psi\rangle$:

$$\langle\sigma|\psi'\rangle = \psi'(\sigma) = \ \langle\sigma|\hat{\mathcal{H}}|\psi\rangle = \sum_{\sigma'} \mathcal{H}^*_{\sigma'\sigma}\psi(\sigma'). \tag{8.4}$$

This gives us the the component σ of $|\psi'\rangle$. The procedure to generate the matrix-vector product is given in the following loop:

$$\hat{\mathcal{H}} = J_1 \sum_{\langle ij\rangle} \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j + J_2 \sum_{\langle\langle ij\rangle\rangle} \hat{\mathbf{S}}_i \cdot \hat{\mathbf{S}}_j \tag{8.5}$$

Algorithm 2 Matrix-vector product *on-the-fly*

```
|ψ'⟩ = complex vector
for |σ⟩ in basis do
    for each j do
        H^(j)_{σ'σ}, |σ'⟩ = Ĥ^(j) |σ⟩
        ψ'(σ)+ = H^(j)*_{σ'σ} × ψ(σ')
    end for
end for
```

9 Block diagonalizing the hamiltonian using translation and spin-flip eigenstates

This is Daniel's take on the math behind reducing the hamiltonian using eigenstates of the translation and parity (spin-flip) operators.[2] I suspect that I am re-deriving standard ideas about characters of group representations, but those concepts have felt rather nebulous to me and this concrete example is very helpful in my understanding.

Let's assume we have a periodic supercell of a hexagonal lattice with M total sites, which thus consists of $N = M/2$ primitive hexagonal cells. Let's further assume that this supercell is constructed so that all the lattice sites can be reached, in order, by consecutive shifts by the primitive lattice vector $\mathbf{a}_1$. Let $|\sigma_i\rangle$ represent simple basis states in the lattice consisting of spin values at each of those sites, and let $\hat{T} = \hat{T}(\mathbf{a}_1)$ be the translation operator corresponding to a translation by $\mathbf{a}_1$. Thus, for example, in an 8-site lattice represented in the notation

$$((a_1, a_2 \ldots, a_N), (b_1, b_2, \ldots, b_N)) \tag{9.1}$$

we could have

$$\hat{T}|\sigma_1\rangle = \hat{T}((+++-), (---+)) = ((-+++), (+---)) \equiv |\sigma_2\rangle. \tag{9.2}$$

Because of the periodic boundary conditions on the supercell, and the fact that we can reach all the sites by successive applications of $\hat{T}$, we must have $\hat{T}^N = 1$, which means that its possible eigenvalues are the N^{th} roots of unity, $e^{-i2\pi q/N}$ for $q \in \{0, 1, \ldots N-1\}$. Because the hamiltonian in Eq. (3.2) commutes with $\hat{T}$, each of the energy eigenstates is simultaneously an eigenstate of $\hat{T}$, so can be labeled by its eigenvalue under $\hat{T}$.

A similar argument can be made for the parity operator, $\hat{P}$, which flips all the spins in any state. Since $\hat{P}$ commutes with both $\hat{\mathcal{H}}$ and $\hat{T}$, the energy eigenstates can also be labeled with their eigenvalue under $\hat{P}$. Since $\hat{P}^2 = 1$, those eigenvalues are simply ± 1.

We're ultimately interested in the energy eigenstates, but we can simplify the problem of finding them by separating the hamiltonian into separate blocks corresponding to the translation and parity eigenvalues. In fact, this is something that we can do without any knowledge of the hamiltonian (though it will only be useful if $\hat{H}$ commutes with $\hat{T}$ and $\hat{P}$).

The eigenstates of $\hat{T}$ are simply the familiar Bloch functions, except here the number of sites is rather small, as opposed to the approximately infinite crystals that are usually considered in textbooks. Starting from any basis state we can construct the set of all translated copies of that state:

$$\left\{|\sigma_i\rangle, \hat{T}|\sigma_i\rangle, \hat{T}^2|\sigma_i\rangle, \ldots, \hat{T}^{N-1}|\sigma_i\rangle\right\} \tag{9.3}$$

where we are guaranteed by the periodic boundary conditions of the supercell that $\hat{T}^N = 1$. (We will come back to the case where $\hat{T}^n = 1$ for some integer $n < N$.) From this set of N original, orthogonal basis states we can construct

[2] Is parity the same thing as spin flip? Sometimes parity is defined as the eigenvalue under inversion symmetry. I guess that could be the case here, since $\mathbf{r} \to -\mathbf{r}$ would take $z \to -z$ and thus flip the z-component of the spin.

N new orthogonal states that are eigenvectors of $\hat{T}$ as was done in the previous section. If we take a (normalized) sum of all N states:

$$|s_0\rangle = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} \hat{T}^m |\sigma_i\rangle = \frac{1}{\sqrt{N}} \left(|\sigma_i\rangle + \hat{T} |\sigma_i\rangle + \hat{T}^2 |\sigma_i\rangle + \cdots + \hat{T}^{N-2} |\sigma_i\rangle + \hat{T}^{N-1} |\sigma_i\rangle \right) \tag{9.4}$$

and then operate on it with $\hat{T}$ and use the property that $\hat{T}^N = 1$,

$$\hat{T} |s_0\rangle = \frac{1}{\sqrt{N}} \left(\hat{T} |\sigma_i\rangle + \hat{T}^2 |\sigma_i\rangle + \hat{T}^3 |\sigma_i\rangle + \cdots + \hat{T}^{N-1} |\sigma_i\rangle + |\sigma_i\rangle \right) = |s_0\rangle \,, \tag{9.5}$$

we see that $|s_0\rangle$ is an eigenstate of $\hat{T}$ with eigenvalue 1. To make an eigenstate with eigenvalue $e^{-i2\pi q/N}$, with integer q less than N, we add in additional phase factors:

$$|s_q\rangle = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} \left(e^{i2\pi q/N} \right)^m \hat{T}^m |\sigma_i\rangle \tag{9.6}$$

$$= \frac{1}{\sqrt{N}} \left(|\sigma_i\rangle + e^{i2\pi q/N} \hat{T} |\sigma_i\rangle + \left(e^{i\pi 2q/N} \right)^2 \hat{T}^2 |\sigma_i\rangle + \cdots + \left(e^{i2\pi q/N} \right)^{N-2} \hat{T}^{N-2} |\sigma_i\rangle + \left(e^{i2\pi q/N} \right)^{N-1} \hat{T}^{N-1} |\sigma_i\rangle \right). \tag{9.7}$$

Operating on $|s_q\rangle$ with $\hat{T}$ yields

$$\hat{T} |s_q\rangle = \frac{1}{\sqrt{N}} \left(\hat{T} |\sigma_i\rangle + e^{i2\pi q/N} \hat{T}^2 |\sigma_i\rangle + \left(e^{i\pi 2q/N} \right)^2 \hat{T}^3 |\sigma_i\rangle + \cdots + \left(e^{i2\pi q/N} \right)^{N-2} \hat{T}^{N-1} |\sigma_i\rangle + \left(e^{i2\pi q/N} \right)^{N-1} |\sigma_i\rangle \right) \tag{9.8}$$

$$= \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} \left(e^{i2\pi q/N} \right)^{m-1} \hat{T}^m |\sigma_i\rangle = \frac{e^{-i2\pi q/N}}{\sqrt{N}} \sum_{m=0}^{N-1} \left(e^{i2\pi q/N} \right)^m \hat{T}^m |\sigma_i\rangle = e^{-i2\pi q/N} |s_q\rangle \,, \tag{9.9}$$

where we make use of the cyclic property of $e^{i2\pi q/N}$. In this way we can turn the original equivalence class of N orthonormal basis vectors in Eq. (9.3) into an orthonormal set of eigenvectors under $\hat{T}$ with eigenvalues $e^{-i2\pi q/N}$ for $q \in \{0, 1, \ldots N-1\}$.

To reconnect with momentum $\mathbf{k}$ in reciprocal space I claim that supercells with the property that all sites can be reached by successive translations by $\mathbf{a}_1$ also have the property that all allowed momenta $\mathbf{k}$ in the primitive BZ can be reached by multiples of one of the supercell reciprocal lattice vectors ($\mathbf{B}_2$ in the specific examples I've worked out): $\mathbf{k} = q\mathbf{B}_2$, with q still an integer less than N. The above eigenvalues then correspond to $e^{-i\mathbf{k}\cdot\mathbf{R}} = e^{-iq\mathbf{B}_2\cdot\mathbf{a}_1} = e^{-iq2\pi/N}$.

The example in Eq. (9.2) for the $M = 8$ site lattice corresponds to one of the $N = \frac{M}{2} = 4$ element "maximal" equivalence classes. However, for $|\sigma_5\rangle = ((+-+-), (-+-+))$ we find

$$\hat{T}|\sigma_5\rangle = \hat{T}((+-+-), (-+-+)) = ((-+-+), (+-+-)) \equiv |\sigma_6\rangle \tag{9.10}$$

$$\hat{T}|\sigma_6\rangle = \hat{T}((-+-+), (+-+-)) = ((+-+-), (-+-+)) = |\sigma_5\rangle \tag{9.11}$$

and thus $|\sigma_5\rangle$ and $|\sigma_6\rangle$ form a 2 element equivalence class. Since $\hat{T}^2 = 1$ on this subspace, the allowed eigenvalues are $e^{-i2\pi q/4}$ for $q \in \{0, 2\}$, namely ± 1, and we can create the corresponding two orthonormal translation eigenstates

$$|s_0\rangle = \frac{1}{\sqrt{2}} \left(|\sigma_5\rangle + |\sigma_6\rangle \right), \qquad |s_2\rangle = \frac{1}{\sqrt{2}} \left(|\sigma_5\rangle - |\sigma_6\rangle \right) \tag{9.12}$$

which is a very simplified case of Eq. (9.6), where the sum runs from $m = 0$ to $m = N_i$ where N_i is the number of states in the translational equivalence class generated by acting $\hat{T}$ on $|\sigma_i\rangle$. Thus N_i is the smallest integer such that $\hat{T}^{N_i}|\sigma_i\rangle = |\sigma_i\rangle$. In the above example $N_i = 2$.

In this way we can build a new set of basis states by forming the N_i translation eigenstates corresponding to every equivalence class generated by different $|\sigma_i\rangle$. Since the hamiltonian commutes with $\hat{T}$, the energy eigenstates can be labeled by their eigenvalues under $\hat{T}$. If we order all the new basis vectors by their $\hat{T}$ eigenvalues then $\hat{\mathcal{H}}$ will be block diagonal, with each block corresponding to a different eigenvalue $e^{i2\pi q/N}$ that can be labeled by q. The current C++ code generates the hamiltonian block corresponding to $q = 0$.

9.1 Spin-flip / Parity

The parity operator can be used to further block diagonalize $\hat{\mathcal{H}}$ in a similar manner, though it is simpler since $\hat{P}^2 = 1$ and thus there are only two cases. In the previous example, since $\hat{P}|\sigma_5\rangle = |\sigma_6\rangle$, the $(|\sigma_5\rangle, |\sigma_6\rangle)$ equivalence class is unchanged under $\hat{P}$, i.e. it is a "singlet" under $\hat{P}$, and so any linear combination (including the translation eigenstates formed above) will be an eigenstate of $\hat{P}$ with eigenvalue $+1$. Thus there is no further reduction one can make in the hamiltonian based on this equivalence class. However, there are also translation equivalence classes that are not closed under parity. The simplest example is the totally antiferromagnetic state,

$$|\sigma_7\rangle = ((++++), (----)). \tag{9.13}$$

This state is a singlet under translations, $\hat{T}|\sigma_7\rangle = |\sigma_7\rangle$, but it transforms under a spin-flip to

$$\hat{P}|\sigma_7\rangle = \hat{P}((++++), (----)) = ((----), (++++)) \equiv |\sigma_8\rangle\,. \tag{9.14}$$

Thus the set $\{|\sigma_7\rangle, |\sigma_8\rangle\}$ is a translation singlet and parity doublet. The corresponding parity eigenstates are

$$|s_{0,\pm}\rangle = \frac{1}{\sqrt{2}}\left(|\sigma_7\rangle \pm |\sigma_8\rangle\right) \tag{9.15}$$

where now I'm labeling the eigenstates $|s_{q,\pm}\rangle$ with the eigenvalues under translation, q, and under parity, ± 1, corresponding to the $\pm$ sign in Eq. (9.15).

The same thing can be done for the earlier example starting with $|\sigma_1\rangle$, which under translations generates an equivalence class $\{|\sigma_1\rangle, |\sigma_2\rangle, |\sigma_3\rangle, |\sigma_4\rangle\}$. Under $\hat{P}$ these states are each mapped to spin-flipped partners $|\sigma_i'\rangle$ that form another 4-element translation equivalence class, which can also be combined to form translation eigenstates with $q = 0, 1, 2, 3$. So altogether these 8 states can be recombined to form 8 new states $|s_{q,\pm}\rangle$ with specific eigenvalues under both $\hat{T}$ and $\hat{P}$, namely $q = 0, 1, 2, 3$ and ± 1.

An $M = 8$ site lattice with $N = M/2 = 4$ primitive unit cells has an $S^z_{\rm tot} = 0$ subspace spanned by 70 "primitive" basis states that are single terms of the form in Eq. (9.1). This is a manageable number and I found it instructive to enumerate all of them and identify their $\hat{T}$ and $\hat{P}$ quantum numbers. The results are collected in Table 2.

The hamiltonian can be put into a block diagonal form with 8 blocks, corresponding to the distinct combinations of q (translation eigenvalue) and p, parity eigenvalue. Table 2 allows us to read off the sizes of these blocks, which are collected in Table 3. For instance, every equivalence class allows for a "trivial" representation, $(q, p) = (0, +)$, yielding 13 different states in that subspace. On the other hand, the top row in Table 2 only admits $q = 0$ states, one with each parity, while for the example in Eq. (9.12) the $q = 0$ state $|s_0\rangle$ has $p = +1$ while the $q = 2$ state $|s_2\rangle$ has $p = -1$.

representative	$\|T\|$	$\|P\|$	# prim. basis
$(----)(++++)$	1	2	2
$(---+)(+++-)$	4	2	8
$(---+)(++-+)$	4	2	8
$(---+)(+-+++)$	4	2	8
$(---+)(-++++)$	4	2	8
$(-+-+)(+--+)$	4	2	8
$(--++)(+-+-)$	4	2	8
$(--++)(--++)$	4	1	4
$(--++)(-++-)$	4	1	4
$(--++)(++--)$	4	1	4
$(--++)(+--+)$	4	1	4
$(-+-+)(-+-+)$	2	1	2
$(-+-+)(+-+-)$	2	1	2

Table 2: Representatives of the 13 equivalence classes under *both* $\hat{T}$ and $\hat{P}$, together with $|T|$, the number of distinct eigenvalues allowed under $\hat{T}$, and $|P|$, the number of eigenvalues allowed under $\hat{P}$, and finally the product, $|T||P|$, which is the total number of primitive states that make up that combined equivalence class. The sum of values in the last column is 70, the total number of primitive basis states needed to span the $S^z_{\rm tot}$ subspace of an $M = 8$ site honeycomb model. For the representatives that have $|P| = 2$, for each q there is a translational eigenstate $p = +1$ and another with $p = -1$. However, for the representatives that have $|P| = 1$ the parity eigenvalue for each translational eigenstate can be *either* $+1$ or -1, but not both.

eigenvalues (q, p)	size of $\hat{\mathcal{H}}_{q,p}$
$(0, +)$	13×13
$(1, +)$	6×6
$(2, +)$	10×10
$(3, +)$	6×6
$(0, -)$	7×7
$(1, -)$	10×10
$(2, -)$	8×8
$(3, -)$	10×10

Table 3: Allowed eigenvalues (q, p) under $\hat{T}$ and $\hat{P}$, respectively, along with the size of the corresponding subspace in the hamiltonian matrix.