RESTRICTED BOLTZMANN MACHINE REPRESENTATION FOR THE GROUNDSTATE AND EXCITED STATES OF KITAEV HONEYCOMB MODEL

Mohammadreza Noormandipour[*,a] , **Youran Sun**[†,b] , **Babak Haghighat**[‡,b]

[a] *DAMTP*, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA, UK

[b] *Yau Mathematical Sciences Center*, Tsinghua University, Beijing, 100084, China

February 26, 2020

ABSTRACT

In this work, the capability of restricted Boltzmann machines (RBMs) to find solutions for the Kitaev honeycomb model is investigated. The measured groundstate (GS) energy of the system is compared and shown to reside within less than 5% error of the analytically derived value of the energy. Furthermore, the possibility of realizing anyons in the RBM is discussed and an algorithm is given to build these anyonic excitations and braid them as a proof of concept for performing quantum gates and doing quantum computation. Moreover, the phase transition of the system is also studied by changing the corresponding hyperparameters of the system in the RBM solution.

Keywords Restricted Boltzmann Machine (RBM) · Feed-Forward Neural Network (FFNN) · Honeycomb Lattice Model · Quantum State Tomography · Topological Phases of Matter · Anyons · Quantum Computing

1 Introduction

Honeycomb model is a 2-dimensional lattice spin system (see Fig. 1) which was first introduced by Kitaev in 2006 [1] and is famous because of the topological quantum order due to a degenerate gapped groundstate which is persistent to local and finite-sized perturbations. The system also supports both abelian and non-abelian topological phases as demonstrated in the original proposal [1]. There is a wide range of applications for this model, from fault-tolerant quantum computation [2] to analytical study of strongly correlated systems [3] and quantum spin liquids [4]. The honeycomb lattice is not a Bravias lattice in its original structure, but it can be considered as a triangular Bravias lattice with a two-spin basis. The direct Bravias lattice and the primitive cells are illustrated in Fig. 1a with the dashed lines. Each primitive cell has a pair of odd and even indexed spins. Just for the sake of simplicity we will denote each cell with the even indexed (empty circles) spins of the system. The primitive vectors of the lattice, $\mathbf{a}_1$ and $\mathbf{a}_2$ are also shown in Fig. 1a (see also Equ. 1). The entire lattice can be tiled and covered with primitive cells using translations composed of different linear combinations of primitive vectors.

$$\mathbf{a}_1 = \sqrt{3}a\mathbf{e}_x \qquad \& \qquad \mathbf{a}_2 = \frac{\sqrt{3}}{2}a(\mathbf{e}_x, \sqrt{3}\mathbf{e}_y) \tag{1}$$

Where a is the lattice constant. The reciprocal lattice for the triangle lattice can be obtained by solving Equ. 2 for basis vectors of the reciprocal lattice, $\mathbf{b}_1$ and $\mathbf{b}_2$.

$$\mathbf{a}_i.\mathbf{b}_j = 2\pi\delta_{ij} \tag{2}$$

---

[*]mrn31@cam.ac.uk

[†]syouran0508@gmail.com

[‡]babakhaghighat@tsinghua.edu.cn

The solution is as below:

$$\mathbf{b}_1 = \frac{2\pi}{\sqrt{3}a}(\mathbf{e}_x - \frac{1}{\sqrt{3}}\mathbf{e}_y) \qquad \& \qquad \mathbf{b}_2 = \frac{4\pi}{3a}\mathbf{e}_y \tag{3}$$

The reciprocal lattice is depicted in Fig.1b.

The Hamiltonian of the Kitaev model is a nearest-neighbor interaction of Pauli matrices on a honeycomb lattice as written in Equ.4, where $r \& r'$ are indices for the nearest neighbour spins. The physics of the system is symmetric under permutation of coupling strengths J_α with $\alpha = x, y, z$ and due to an-isotropic interaction the model is a fraustrated spin system, because a spin cannot satisfy conflicting demands of orientation from its three neighboring sites [1].

$$H = -\sum_\alpha J_\alpha \sum_{\alpha-bonds} \sigma_r^\alpha \sigma_{r'}^\alpha \tag{4}$$

In this paper we apply the approach developed in [5] to the Honeycomb model. In sec.2, we briefly review the analytical solution of the model. In sec.3 the mapping of the spin model to an RBM architecture is explained. In particular, we demonstrate how to calculate the groundstate and excited states of the Honeycomb model using machine learning techniques. To this end we employ the `NetKet` environment [6] to train a restricted Boltzmann machine (RBM) in order to find a groundstate via gradient decent. In sec.4 realization of vortices is discussed from both analytical and RBM language points of view. In brief, a vortex pair can be created by adjusting the parameters of the RBM and subsequent braiding can be achieved by further adjustments of parameters. Sec.5 is dedicated to results and relevant discussions and finally we conclude in sec.6. An important application of the techniques developed in thi is the reconstruction of a given quantum state from simple measurements known as quantum state tomography [7].
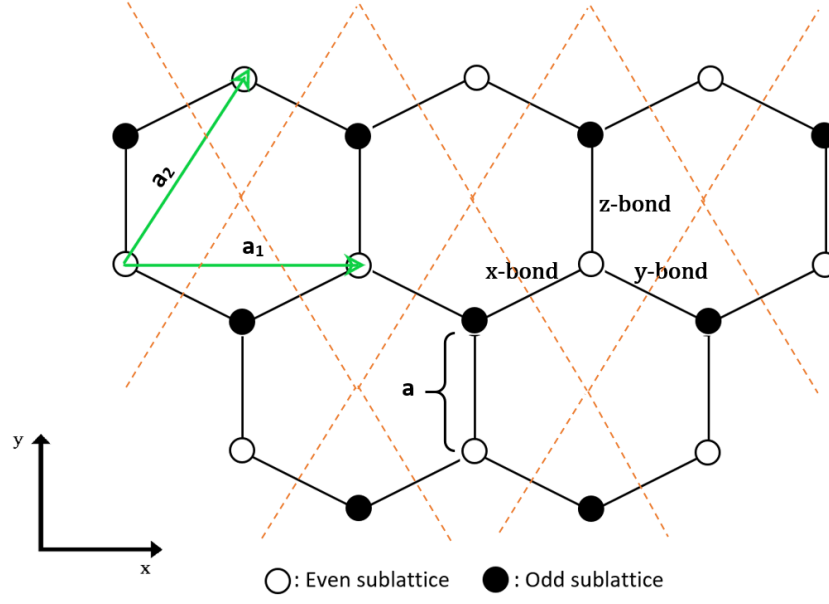
2 Solution of the Model

In this section an exact solution of the system is provided and an explicit form of the groundstate energy is achieved. First of all, we Fermionize the Hamiltonian by performing a one dimensional Jordan-Wigner transformation [8] defined in Equ.5. A deformation of the hexagonal lattice to a brick-wall lattice (see Fig.2) clarifies the mechanism of the Jordan-Wigner transformation and why it is one dimensional [9]. In the brick-wall lattice, each lattice site r is denoted by the coordinates (i, j).
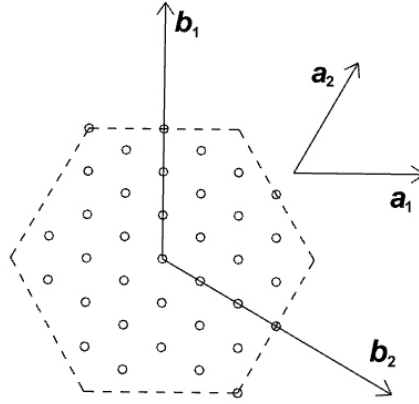
$$\sigma_{i,j}^+ = 2[\prod_{j'<j}\prod_{i'}\sigma_{i',j'}^z][\prod_{i'<i}\sigma_{i',j}^z]a_{i,j}^\dagger$$
$$\sigma_{i,j}^- = 2[\prod_{j'<j}\prod_{i'}\sigma_{i',j'}^z][\prod_{i'<i}\sigma_{i',j}^z]a_{i,j} \tag{5}$$
$$\sigma_{i,j}^z = 2a_{i,j}^\dagger a_{i,j} - 1$$

This transformation maps the Hilbert space of spins to the Hilbert space of spinless complex fermions. Using the fact that $\sigma^\pm = \sigma^x \pm i\sigma^y$ one can expand the Hamiltonian in Equ.4 into three terms and re-write them as below [10].

$$\sigma_{i,j}^x \sigma_{i+1,j}^x = \prod_{i'<i}\sigma_{i',j}^z(a_{i,j}^\dagger + a_{i,j}) \prod_{i'<i+1}\sigma_{i',j}^z(a_{i+1,j}^\dagger + a_{i+1,j})$$
$$= (a_{i,j}^\dagger + a_{i,j})\sigma_{i,j}^z(a_{i+1,j}^\dagger + a_{i+1,j})$$
$$= -(a_{i,j}^\dagger - a_{i,j})(a_{i+1,j}^\dagger + a_{i+1,j})$$
$$\sigma_{i,j}^y \sigma_{i+1,j}^y = -\prod_{i'<i-1}\sigma_{i',j}^z(a_{i-1,j}^\dagger - a_{i-1,j})\prod_{i'<i}\sigma_{i',j}^z(a_{i,j}^\dagger - a_{i,j}) \tag{6}$$
$$= -(a_{i-1,j}^\dagger - a_{i-1,j})\sigma_{i-1,j}^z(a_{i,j}^\dagger - a_{i,j})$$
$$= (a_{i-1,j}^\dagger + a_{i-1,j})(a_{i,j}^\dagger - a_{i,j})$$
$$\sigma_{i,j}^z \sigma_{i,j+1}^z = (2a_{i,j}^\dagger a_{i,j} - 1)(2a_{i,j+1}^\dagger a_{i,j+1} - 1)$$

(a)

(b)

Figure 1: (a) The honeycomb lattice in position space and its primitive vectors and primitive cell. (b) First Brillouin zone and primitive vectors of reciprocal lattice.

Therefore, the Hamiltonian transforms to the one in Equ.7.

$$
\begin{aligned}
H = & + J_x \sum_{x-links} (a_{i,j}^\dagger - a_{i,j})(a_{i+1,j}^\dagger + a_{i+1,j}) \\
& - J_y \sum_{y-links} (a_{i-1,j}^\dagger + a_{i-1,j})(a_{i,j}^\dagger - a_{i,j}) \\
& - J_z \sum_{z-links} (2a_{i,j}^\dagger a_{i,j} - 1)(2a_{i,j+1}^\dagger a_{i,j+1} - 1)
\end{aligned}
\tag{7}
$$

3

Figure 2: Brick-wall lattice and the Jordan-Wigner transformation path depicted with the dashed arrows. The $\mathbf{r}_1$ and $\mathbf{r}_2$ are the primitive vectors for this lattice.

Figure 3: Kitaev Honeycomb lattice model on a torus: The generators of the lattice in position space are named $\mathbf{e}_1$ and $\mathbf{e}_2$, as shown in the image above. Hence, the lattice points are $\mathbf{L} = n_1\mathbf{e}_1 + n_2\mathbf{e}_2$ with $n_1, n_2 \in \mathbb{Z}$ and the above lattice is drawn for $N_1 = N_2 = 4$ with toric boundary conditions.

The J_x and J_y terms are quadratic interactions in spinless fermions and are easy to solve, however, the J_z term is a product of number density operators and can be further simplified by introducing the Majorana operators in Equ. 8 [10]. As we will see, this simplification can be done due to the presence of the conserved quantity called plaquette operator B_p [10].

$$
\begin{aligned}
c_{i,j} = i(a^\dagger_{i,j} - a_{i,j}) \ , \ d_{i,j} = a^\dagger_{i,j} + a_{i,j} \ , \ \ for \ \ i+j = even \equiv \circ \\
c_{i,j} = a^\dagger_{i,j} + a_{i,j} \ \ , \ d_{i,j} = i(a^\dagger_{i,j} - a_{i,j}) \ , \ \ for \ \ i+j = odd \equiv \bullet
\end{aligned}
\tag{8}
$$

These operators have the following commutation relations:

$$
\begin{aligned}
c^2_{i,j} = d^2_{i,j} = 1 \\
\{c_{i,j}, c_{i',j'}\} = \{d_{i,j}, d_{i',j'}\} = 2\delta_{ii'}\delta_{jj'} \\
\{c_{i,j}, d_{i',j'}\} = 0
\end{aligned}
\tag{9}
$$

Then, the J_z term can be rewritten using the Majorana operators:

$$
\begin{aligned}
\sigma^z_{i,j}\sigma^z_{i,j+1} &= (2a^\dagger_{i,j}a_{i,j} - 1)(2a^\dagger_{i,j+1}a_{i,j+1} - 1) \\
&= i(id_{i,j+1}d_{i,j})c_{i,j+1}c_{i,j}
\end{aligned}
\tag{10}
$$

Finally, using the circle indices for the odd and even lattice sites as defined in Equ. 8 the Hamiltonian transforms to the expression in Equ. 11.

$$
\begin{aligned}
H = &- iJ_x \sum_{x-links} c_\circ c_\bullet \\
&+ iJ_y \sum_{y-links} c_\bullet c_\circ \\
&- iJ_z \sum_{z-links} (id_\bullet d_\circ)c_\bullet c_\circ
\end{aligned}
\tag{11}
$$

If we write the Hamiltonian as a sum over unit cells we have:

$$
H = i\sum_{\mathbf{r}}[J_x c_{\bullet,\mathbf{r}}c_{\circ,\mathbf{r}+\mathbf{r_1}} + J_y c_{\bullet,\mathbf{r}}c_{\circ,\mathbf{r}+\mathbf{r_2}} - J_z(id_{\bullet,\mathbf{r}}d_{\circ,\mathbf{r}})c_{\bullet,\mathbf{r}}c_{\circ,\mathbf{r}}]
\tag{12}
$$

where $\mathbf{r}$ is the position vector of z-bonds or unit cells (see Fig. 2). It makes no difference in physics if we set $\mathbf{r}$ on every point along the z-bond, whether it is on even site or odd site or some where in the middle of them, that's a matter of translation! What will be important in what follows, is that now since we have grouped a pair of even and odd sites as an unit cell, to evaluate the summation over the unit cells, it is sufficient to just run over the even or odd sites. Furthermore, the $\alpha_{\mathbf{r}} = (id_{\bullet,\mathbf{r}}d_{\circ,\mathbf{r}})$ operators are defined on each z-bond of the lattice (labled by $\mathbf{r}$) and they commute with the Hamiltonian and are good quantum numbers. Moreover, it can easily be shown that each plaquette operator can be written as:

$$
B_p = \sigma^y_1\sigma^z_2\sigma^x_3\sigma^y_4\sigma^z_5\sigma^x_6 = \alpha_{61}\alpha_{43}
\tag{13}
$$

5

On the other hand, from the Lieb's theorem [11] we know that the groundstate manifold is obtained by setting $B_p = 1, \forall p$. Thus the uniform choice of $\alpha_{\mathbf{r}} = 1, \forall \mathbf{r}$ corresponds to a vortex-free sector, nevertheless all configurations leading to the same sector are equivalent.

We also introduce a Dirac fermion on each z-link using the Majorana operators:

$$
\begin{aligned}
d_{\mathbf{r}} &= \frac{1}{2}(c_{\bullet,\mathbf{r}} - ic_{\circ,\mathbf{r}}) \\
d_{\mathbf{r}}^{\dagger} &= \frac{1}{2}(c_{\bullet,\mathbf{r}} + ic_{\circ,\mathbf{r}})
\end{aligned}
\tag{14}
$$

Using the inverse transformation we can rewrite the Hamiltonian:

$$
\begin{aligned}
H = \sum_{\mathbf{r}} [&J_x(d_{\mathbf{r}}^{\dagger} + d_{\mathbf{r}})(d_{\mathbf{r}+\mathbf{r}_1}^{\dagger} - d_{\mathbf{r}+\mathbf{r}_1}) \\
&+ J_y(d_{\mathbf{r}}^{\dagger} + d_{\mathbf{r}})(d_{\mathbf{r}+\mathbf{r}_2}^{\dagger} - d_{\mathbf{r}+\mathbf{r}_2}) \\
&+ J_z\alpha_{\mathbf{r}}(2d_{\mathbf{r}}^{\dagger}d_{\mathbf{r}} - 1)]
\end{aligned}
\tag{15}
$$

The Hamiltonian is translational invariant and can be transformed to momentum space in order to be diagonalized. We define the Fourier transf for the Dirac fermion through:

$$
\begin{aligned}
d_{\mathbf{r}} &= \frac{1}{\sqrt{N}} \sum_{\mathbf{k}} e^{+i\mathbf{k}.\mathbf{r}} d_{\mathbf{k}} \\
d_{\mathbf{r}}^{\dagger} &= \frac{1}{\sqrt{N}} \sum_{\mathbf{k}} e^{-i\mathbf{k}.\mathbf{r}} d_{\mathbf{k}}^{\dagger}
\end{aligned}
\tag{16}
$$

Setting $\mathbf{k} \to \text{-}\mathbf{k}$ in $d_{\mathbf{r}}^{\dagger}$ simplifies the calculations. We then have:

$$
\begin{aligned}
\mathbf{X}: \quad &J_x\frac{1}{N}\sum_{\mathbf{r}}\sum_{\mathbf{k},\mathbf{k}'} e^{+i\mathbf{k}.\mathbf{r}}e^{+i\mathbf{k}'.(\mathbf{r}+\mathbf{r}_1)}(d_{-\mathbf{k}}^{\dagger} + d_{\mathbf{k}})(d_{-\mathbf{k}'}^{\dagger} - d_{\mathbf{k}'}) \\
&= J_x\sum_{\mathbf{k}}[-2cos(k_1)d_{\mathbf{k}}^{\dagger}d_{\mathbf{k}} + isin(k_1)(d_{\mathbf{k}}^{\dagger}d_{-\mathbf{k}}^{\dagger} - h.c.)] \\
\mathbf{Y}: \quad &J_y\frac{1}{N}\sum_{\mathbf{r}}\sum_{\mathbf{k},\mathbf{k}'} e^{+i\mathbf{k}.\mathbf{r}}e^{+i\mathbf{k}'.(\mathbf{r}+\mathbf{r}_2)}(d_{-\mathbf{k}}^{\dagger} + d_{\mathbf{k}})(d_{-\mathbf{k}'}^{\dagger} - d_{\mathbf{k}'}) \\
&= J_y\sum_{\mathbf{k}}[-2cos(k_2)d_{\mathbf{k}}^{\dagger}d_{\mathbf{k}} + isin(k_2)(d_{\mathbf{k}}^{\dagger}d_{-\mathbf{k}}^{\dagger} - h.c.)] \\
\mathbf{Z}: \quad &J_z\frac{1}{N}\sum_{\mathbf{r}}\sum_{\mathbf{k},\mathbf{k}'} e^{+i\mathbf{k}.\mathbf{r}}e^{+i\mathbf{k}'.\mathbf{r}}(2d_{-\mathbf{k}}^{\dagger}d_{\mathbf{k}'}^{\dagger} - 1) \\
&= J_z\sum_{\mathbf{k}} 2d_{\mathbf{k}}^{\dagger}d_{\mathbf{k}} - J_zN.
\end{aligned}
\tag{17}
$$

For the above equation we used the orthogonality relation in Fourier transformation:

$$
\sum_{\mathbf{r}} e^{i(\mathbf{k}+\mathbf{k}').\mathbf{r}} = N\delta(\mathbf{k} + \mathbf{k}')
\tag{18}
$$

Using all previous transformations we obtain a Hamiltonian which is quadratic in the Dirac fermions living on each of the unit cells along the z-links:

$$H = \sum_{\mathbf{k}} [\epsilon_{\mathbf{k}} d_{\mathbf{k}}^{\dagger} d_{\mathbf{k}} + \frac{1}{2} (i\Delta_{\mathbf{k}} d_{\mathbf{k}}^{\dagger} d_{-\mathbf{k}}^{\dagger} - i\Delta_{\mathbf{k}} d_{-\mathbf{k}} d_{\mathbf{k}})] - J_z N$$
$$\epsilon_{\mathbf{k}} = 2[J_z - J_x cos(k_1) - J_y cos(k_2)]$$
$$\Delta_{\mathbf{k}} = 2[J_x sin(k_1) + J_y sin(k_2)]$$

(19)

By applying a unitary Bogoliubov transformation, we diagonalize the Hamiltonian which can be written as:

$$H = \sum_{\mathbf{k}} \frac{1}{2} \begin{pmatrix} \gamma_{\mathbf{k}}^{\dagger} & \gamma_{\text{-}\mathbf{k}} \end{pmatrix} \begin{pmatrix} E_{\mathbf{k}} & 0 \\ 0 & -E_{\mathbf{k}} \end{pmatrix} \begin{pmatrix} \gamma_{\mathbf{k}} \\ \gamma_{\text{-}\mathbf{k}}^{\dagger} \end{pmatrix}$$
$$= \sum_{\mathbf{k}} E_{\mathbf{k}} (\gamma_{\mathbf{k}}^{\dagger} \gamma_{\mathbf{k}} - \frac{1}{2}).$$

(20)

with a groundstate energy of $E_{GS} = -\frac{E_{\mathbf{k}}}{2}$.

3 Restricted Boltzmann Machine Representation

With the ever growing applications of neural networks in sciences and the emergent new technologies to deploy and build physical neural networks, this is the right time to investigate potential applications of them in condensed matter systems. Recently, a new approach has been proposed for simulating topological quantum states using neural networks [5]. In this section we use the same approach to map the Honeycomb Kitaev model to a restricted Boltzmann machine (RBM). There are many reasons why an RBM is chosen. This particular architecture has proved to be effective in many tasks such as dimensional reduction, classification, regression, collaborative filtering, feature learning and topic modeling and moreover a general theorem shows that RBMs are universal approximators of discrete distributions [12].

Having a set of spins on a lattice, $\Xi = (\sigma_1, \sigma_2, \cdots \sigma_N)$ (in our case the Jordan-Wigner chain of spins), our goal is to use an RBM to reduce the dimensionality of the Hilbert space and estimate the energy of the system in both groundstate and excited states and classify different topological phases of the model. An RBM is a short range feed-forward neural network with two layers. The first layer (visible layer) has N nodes which are representing the physical spins in the Hamiltonian and the second layer (hidden layer) has M binary valued nodes (architecture of the network is shown in Fig.4).
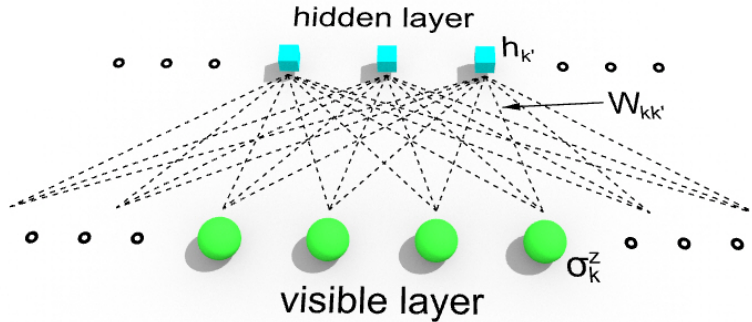

Figure 4: Fully connected Restricted Boltzmann machine architecture.

The quantum state of the Honeycomb model, up to an irrelevant normalization factor, can be written as

$$|\Phi\rangle = \sum_{\Xi} \Phi_M(\Xi; \Omega) |\Xi\rangle \tag{21}$$

where

$$
\begin{aligned}
\Phi_M(\Xi; \Omega) &= \sum_{\{h_k\}} e^{\sum_k a_k \sigma_k^z + \sum_{k'} b_{k'} h_{k'} + \sum_{kk'} W_{kk'} h_k \sigma_{k'}^z} \\
&= e^{\sum_k a_k \sigma_k^z} \times \prod_{k'} \cosh\left(\sum_k W_{kk'} \sigma_k^z + b_{k'}\right)
\end{aligned}
\tag{22}
$$

To obtain the second equality we used the values for $\{h_k\} = \{-1, 1\}^M$ which is the set of possible configurations of hidden layer nodes and the $\Omega = (a_k, b_{k'}, W_{kk'})$ is the set of weights and biases of the RBM which should be trained in such a way that the final RBM state represents the desired quantum state of the model (i.e. groundstate or the excited states). The combined number of weight and bias parameters and nodes is polynomial in system size and computationally feasible.

Mapping of the model to RBM was done using the machinery already developed in the `NetKet` software package [6]. In order to train the network, the parameter space of the network was sampled using the Metropolis algorithm and the optimization iterations are based on stochastic gradient descent algorithm.

4 Braiding

In this section we want to explore the possibility to create anionic excitations in the RBM representation and describe anyon braiding. Such a representation is important for various reasons, in particular for performing quantum computation [13].

First of all, let us describe the creation of these quasi-particles by starting with a groundstate sector and then subsequently changing the eigenvalue of the B_p operators from +1 to -1 locally and in the region that we want to realize the vortices in the system. In an arbitrary configuration of spins (not necessarily in the groundstate sector) it is easy to prove that $\prod B_p = +1, \forall p$. Hence, we can just build the vortices in pairs. For example, if you apply the operator $\hat{O}_1$ to the groundstate, it produces two vortices in plaquettes 1 and 2 (see Fig.5).

Do we need a reference here or are these ideas completely new?

$$\hat{O}_1 = \exp\left(-i\frac{\pi}{2}\hat{\sigma}_a^z\right) \tag{23}$$

Another possibility is to apply the following operator

$$\hat{O}_2 = \exp\left(-i\frac{\pi}{2}\hat{\sigma}_a^x\right) \exp\left(-i\frac{\pi}{2}\hat{\sigma}_b^y\right), \tag{24}$$

which produces two vortices along the $\hat{z}$ direction in plaquettes 3 and 4.

The reason why these operators can produce vortices in the system can be explained by re-writing the Hamiltonian of the system in terms of Majorana fermions. Kitaev originally solved the Hamiltonian through this approach. The disadvantage of this approach in comparison to what was done above is that when the Hamiltonian is mapped to Majorana fermions, there are unphysical states in the system which need to be projected out. This will be clear as we go along.

We assume that there are two fermionic modes living on each lattice site corresponding to the four creation and annihilation operators $a_{m,i}^\dagger$ and $a_{m,i}$, where $m \in \{1, 2\}$ is the index for the modes and i is the index for the lattice sites.
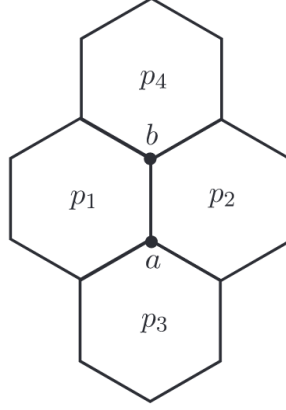
Figure 5: Caption

Here unlike before, we use one index for lattice site to avoid unnecessary complications. Then we will separate the imaginary and real parts of these operators, to define the Majorana fermions as below

$$
\begin{aligned}
c_i &= a_{1,i} + a_{1,i}^\dagger \\
b_i^x &= i(a_{1,i}^\dagger - a_{1,i}) \\
b_i^y &= a_{2,i} + a_{2,i}^\dagger \\
b_i^z &= i(a_{2,i}^\dagger - a_{2,i})
\end{aligned}
\tag{25}
$$

Notice that the spins have a two dimensional space (being up or down) and now that we are representing them with four Majorana modes (two complex fermionic modes), we need to project out the unphysical states. The Fock space of the complex fermionic modes can be represented as $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$. We make the following correspondence between the states of spins and fermions

$$
\begin{aligned}
|\uparrow\rangle &= |00\rangle \\
|\downarrow\rangle &= |11\rangle
\end{aligned}
\tag{26}
$$

We can define the projector P_i on site i to do this job for us

$$
P_i = \frac{1 + D_i}{2} \quad \text{where} \quad D_i = (1 - 2a_{1,i}^\dagger a_{1,i})(1 - 2a_{2,i}^\dagger a_{2,i}) = b_i^x b_i^y b_i^z c_i .
\tag{27}
$$

One can easily show that the relation between Majorana operators and the original Pauli operators is as follows

$$
\sigma_i^\alpha = i b_i^\alpha c_i \quad \text{for} \quad \alpha \in \{x, y, z\} .
\tag{28}
$$

In fact, by the above projection, we satisfied the extra condition coming from the algebra of Pauli matrices, $-i\sigma_i^x \sigma_i^y \sigma_i^z = b_i^x b_i^y b_i^z c_i = \mathbb{1}$. Therefore, the eigenvalue of the projector P_i for physical states is 1 and for unphysical states is 0.

Using Equ.28 the Hamiltonian can be written in terms of Majorana operators as

$$
H = \frac{i}{2} \sum_{i,j} A_{ij} c_i c_j \quad \text{where} \quad A_{ij} = J_{ij} u_{ij} \quad \text{and} \quad u_{ij} = i b_i^\alpha b_j^\alpha \quad \text{with} \quad \alpha \in \{x, y, z\} .
\tag{29}
$$

9

The u_{ij} are antisymmetric Hermitian link operators with eigenvalues ± 1

$$u_{ij} = -u_{ji}, \ \ u_{ij}^2 = 1, \ \ , u_{ij}^\dagger = u_{ij} \tag{30}$$

The link operators commute with the Hamiltonian, $[H, u_{ij}] = 0$, so they are local symmetries. In this form, the Hamiltonian is representing a tight binding model of free Majorana fermions hopping along the lattice, with tunneling couplings that depend on the eigenvalues of link operators which can be thought of as a classical $\mathbb{Z}_2$ gauge field. One can assign a configuration $\{u\}$ to all link operators and diagonalize the resulting quadratic Hamiltonian in Majorana operators directly and obtain the spectrum and groundstate energy for $H\{u\}$. But which configuration $\{u\}$ corresponds to the global minimum of the energy? This question has been answered by Lieb (1994) [14]. The groundstate lies in the sector with $B_p = 1 \ \forall p$ and since the B_p operators are the only gauge invariant objects, every gauge fixing choice $\{u\}$ which leads to this particular configuration for B_p is equivalent. Therefore, our goal is to find a configuration $\{u\}$ which leads to $B_p = 1 \ \forall p$.

Next, we define the vortex operator W_p as

$$W_p = \prod_{(i,j) \in \partial P} u_{ij} \ \ \text{where} \ \ \begin{cases} i \in & \text{even sublattice} \\ j \in & \text{odd sublattice} \end{cases} . \tag{31}$$

The vortex operator can be simplified as below

$$
\begin{aligned}
W_p &= u_{12}^x u_{32}^y u_{34}^z u_{54}^x u_{56}^y u_{16}^z \\
&= -(i)^6 c_1^x c_2^x c_2^y c_3^y c_3^z c_4^z c_4^x c_5^x c_5^y c_6^y c_6^z c_1^z \\
&= \sigma_1^y D_1 \sigma_2^z D_2 \sigma_3^x D_3 \sigma_4^y D_4 \sigma_5^z D_5 \sigma_6^x D_6 ,
\end{aligned}
\tag{32}
$$

where we used $i c_j^x c_j^y = -\sigma_j^z D_j$ and cyclic permutations. Considering the fact that $[W_p, H] = [W_p, D_i] = 0$, in the physical subspace where $D_j = 1$ we recover $W_p = B_p$. Therefore, the uniform choice of $u_{ij} = 1$ for every link will lead us to the groundstate in the extended (not yet projected) space with $B_p = 1$. In order to find the physical groundstate, we apply the global projector $\mathbf{D}$ to project out the unphysical states

$$|\psi_w\rangle_{physical} = \mathbf{D} |\psi_u\rangle_{extended}, \ \ \text{where} \ \ \mathbf{D} = \prod_{i=1}^{N} P_i \tag{33}$$

Now we are at a point to explain in detail why the operators $\hat{O}_1$ and $\hat{O}_2$ can create vortices in the system. As mentioned earlier, if we flip the sign of eigenvalue of vortex operator at p, a vortex will be created in the system. Looking back at Equ.31, we see that the vortex configuration $\{W_p\}$ is created by fixing the gauge at every link $\{u_{ij}\}$. Therefore, by changing the link operators, we can move between vortex sectors. Starting from a vortex-free sector with all link operators being $+1$, applying the $\hat{O}_1$ on site a will leave the σ_a^z untouched while flipping the σ_a^x and σ_a^y. Consequently, this will change the sign of the eigenvalue of the u_{ij} for x-link and y-link attached to site a, leaving the system with $B_{p_1} = -1$ and $B_{p_2} = -1$. We can also move the vortices around in the system, by consecutively applying these operators on the particular sites in a path for which we want to move the vortices along. As an example, the red path shown in the Fig.6 corresponds to the operator

$$\hat{O} = \exp\left(-i\frac{\pi}{2}\sigma_i^z\right) \exp\left(-i\frac{\pi}{2}\sigma_j^x\right) \tag{34}$$

In general, we can define string operators to move the vortices anywhere in the system along the string, similar to the example above. To create such a string operator is straightforward now. If the string passes through a link, we need to
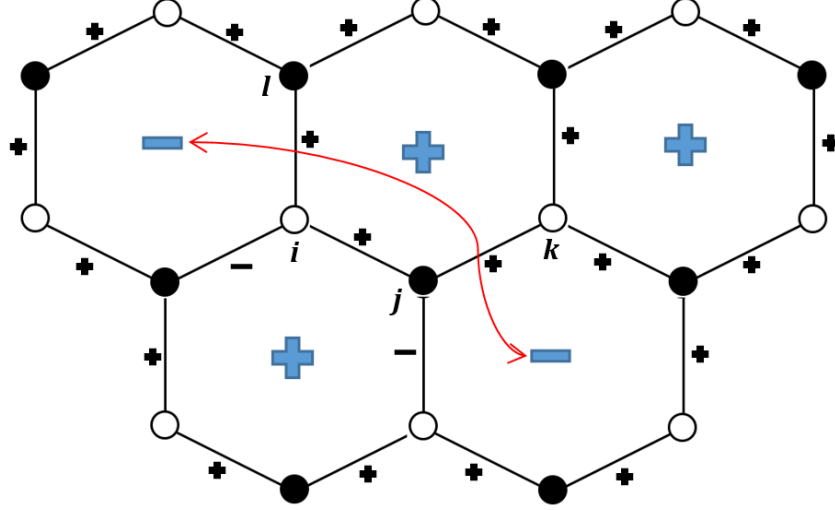
Figure 6: Two vortices connected by a string operator passing through several links. The white dots denote even lattice sites while the black dots are odd ones. By convention, the string operator is built up from operators acting on even sites.

apply an operator on one of the sites connected to that $\alpha - link$ which carries σ^α in the exponent. Before continuing, we need to make a convention. A sharp reader might point out that applying the operator $\hat{O}' = \exp\left(-i\frac{\pi}{2}\sigma_l^z\right)\exp\left(-i\frac{\pi}{2}\sigma_k^x\right)$ instead of $\hat{O}$ creates the same pair of vortices in the system. For consistency reasons, we therefore use the convention that if a string is passing through a link, the corresponding string operator is built up by picking the even site on the link. Having said all of this, a generic string operator is defined as

$$\hat{S} = \prod_{\alpha-links} \exp\left(-i\frac{\pi}{2}\hat{\sigma}_\circ^\alpha\right).$$
(35)

This string operator will create two vortices at the ending points of the string which passes through the $\alpha - links$. The manipulation of link operators using these string operators is practically equivalent to changing the sign of the couplings J_{ij} corresponding to the link u_{ij} in the definition of the A_{ij} (see Equ.29). Therefore, in the RBM language one can simply redefine the Hamiltonian with the desired signs of J_{ij} for a particular vortex sector and then try to find the RBM representation for this new Hamiltonian through the approach explained in section 3. The only practical disadvantage of this vortex realization approach is that it is computationally expensive as the RBM should be trained again from scratch to find the representation for the excited state. An ideal approach is that, given the representation for groundstate, we would be able to directly and without further optimization steps find the representation for the excited state.

In principle, it is possible to directly manipulate the weight and bias parameters of the RBM, $\Omega = (a_k, b_{k'}, W_{kk'})$ to realize the vortices and find the excited state representation. Looking at Equ.22, we notice that there is a symmetry between the eigenvalues of physical spins and the value of parameters of the RBM. In other words, flipping a particular spin in site k is equivalent to flipping the sign of the RBM parameter sitting next to the σ_k^α in Equ.22 and/or adding a phase to it. In general, for a string operator $\hat{S}_a = \exp\left(-i\frac{\pi}{2}\hat{\sigma}_a^\alpha\right)$ we have:

$$\hat{S}\,|\Phi\rangle = (-i\hat{\sigma}_a^\alpha)\sum_{\Xi}\Phi_M(\Xi;\Omega)|\Xi\rangle$$

$$= \sum_{\Xi} e^{[(-1)^{\delta(\alpha-z)-1}a_a-i\frac{\pi}{2}\sigma_a^z]\sigma_a^z+\sum_{k\neq a}a_k\sigma_k^z} \times$$
(36)

$$\prod_{k'}\cosh\left((-1)^{\delta(\alpha-z)-1}W_{ak'}\sigma_a^z + \sum_{k\neq a}W_{kk'}\sigma_k^z + b_{k'}\right)\hat{\sigma}_a^\alpha\,|\Xi\rangle$$

11

Therefore, applying $\hat{S}_a$ on RBM state, is equivalent to multiplying the bias parameter a_a and weight parameters $W_{ak'}$ $\forall k'$ by $(-1)^{\delta(\alpha-z)-1}$ and then manually adding a $(-i\frac{\pi}{2}\sigma_a^z)$ phase angle to a_a, where $\sigma_a^z \in \{-1,+1\}$ is the eigenvalue of the z component of the spin on site a. The presence of the $(-1)^{\delta(\alpha-z)-1}$ factor is to ensure that when $\alpha = x$ or y, the $\hat{\sigma}_a^z$ eigenvalue is multiplied by a minus sign.

5 Results and Discussion

5.1 Groundstate

In this section, we present the results for the groundstate energy estimation of different system sizes for the proposed mapping in the section 3. These results are also compared to the exact analytical expression given in Equ.20. The absolute value of the energies is plotted in Fig.7.

The RBM has the simple fully-connected and two-layered architecture with an optimized value of $\alpha = \frac{num_h}{num_v} = 2$ which is the ratio of number of hidden nodes to the number of visible nodes. For lower values of α the network does not give an accurate estimation of energies, while the larger values cause over-fitting and again inaccurate results. For training, the configuration space of the network has been sampled for hundreds of times, depending on the lattice size, using Metropolis algorithm. Moreover, the stochastic gradient descent algorithm has been used for optimization for enough number of iterations such that the network converge to a low enough minima of the energy landscape.

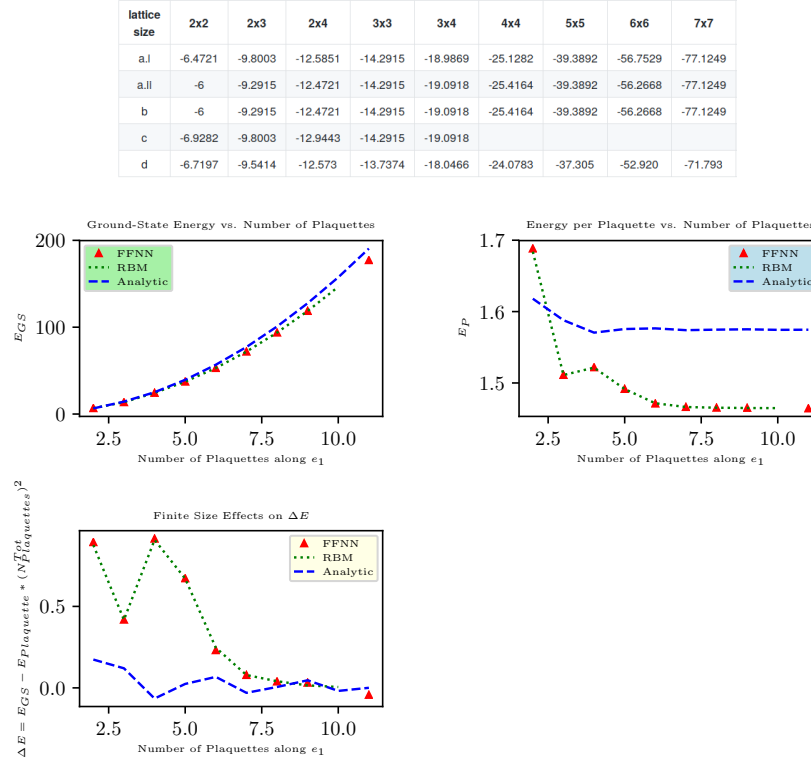| lattice size	2x2	2x3	2x4	3x3	3x4	4x4	5x5	6x6	7x7
a.I	-6.4721	-9.8003	-12.5851	-14.2915	-18.9869	-25.1282	-39.3892	-56.7529	-77.1249
a.II	-6	-9.2915	-12.4721	-14.2915	-19.0918	-25.4164	-39.3892	-56.2668	-77.1249
b	-6	-9.2915	-12.4721	-14.2915	-19.0918	-25.4164	-39.3892	-56.2668	-77.1249
c	-6.9282	-9.8003	-12.9443	-14.2915	-19.0918				
d	-6.7197	-9.5414	-12.573	-13.7374	-18.0466	-24.0783	-37.305	-52.920	-71.793

Figure 7: Numerical results for groundstate energy calculation using Restricted Boltzmann Machines (RBM) and Feed-Forward Neural Networks (FFNN) compared to the analytically obtained groundstate energy using Equ.19.

Based on the results obtained for the energy of the state of system, training of the network was tested and optimized by changing the learning rate for different training phases and iteration numbers. However, the outcome was not considerably changed and improved and it seems that the above results are the most accurate results one can get for this particular architecture of network.

In order to check the effect of relative number of hidden and visible nodes on the outcome of the network, different values for α were tested, keeping other parameters fixed. The α parameter was changed just by changing the number of hidden nodes for a fixed system size (and hence fixed visible layer size) and the results are outlined in the table below,

including the number of free parameters of the network and the time needed for the network to be trained. The exact analytical energy for the selected spin system is -39.3892.

alpha	num_nh	num_para	energy	time(s)	notes
0.1	5	305	-27.7808(59)	171.6	
0.3	15	815	-33.4312(37)	455.0	
0.5	25	1325	-36.6024(22)	711.9	
1.0	50	2600	-37.2464(13)	1549	
1.5	75	3875	-37.1235(15)	2764	
2.0	100	5150	-37.3054(14)	5611	
3.0	150	7700	-37.3037(14)	8415	
4.0	200	10250	-37.1007(16)	11204	
6.0	300	15350	-37.2989(14)	18167	
8.0	400	20450	-37.1710(15)	26000	
10.0	500	25550	-37.1946(15)	32056	

Figure 8: Numerical results for the number of hidden nodes and its influence on energy calculations. $\alpha = \frac{num_h}{num_v}$.

– 2 is the most suitable value for alpha. If it is too low, the fitting performace is not good; too high overfit will happen.
– The training time is approximately proportional to the number of parameters. For each additional parameter, the training time is increased by about 1 second.
– The result of RBM is still far away from the analytic result.

The performance of FFNN

Below is the result of FFNN with one layer. As the structure of ffnn is similar to that of a no-visible-bias RBM(reffered as novb RBM below), the performance of one-layer FFNN is compared to that of no-visible-bias RBM.

alpha	num_nh	num_para	ffnn_energy	novb_rbm_energy	ffnn_time(s)	novb_rbm_time	notes
1.0	50	2550	-37.2401(15)	-37.1532(15)	2783	2754	
2.0	100	5100	-37.2958(14)	-36.8954(18)	4591	5476	
3.0	150	7650	-37.3007(14)	-37.2191(15)	6889	8331	

Figure 9: FFNN analysis.

– FFNN trains faster than RBM and performs better than RBM with no bias.
– But FFNN's result again far away from the analytic one.

Deep FFNNs with 2 and 3 layers are also tried. But some bugs seem happen, the energies stay at -25. It will be studied later.

5.2 Excited States: Realization of Anyons

Create Vortex and Measure its energy

There are three theoretically equivalent method to create vortices

(a) modifing parameters of RBM
(b) creating a auxiliary Hamiltonian and train again

13

(c) transforming into auxiliary fermion and fixing u_{ij}s (Kitaev and Pachos)

I will first triple check using 3x3 lattice. The numeric result of gs energy of 3x3 lattice is -13.7166(2), the theoretical gs energy is -14.2915. The results is as the table below.

spins be flipped	energy of a	energy of b	energy of c
8xz	-11.7335(5)	-12.1070(51)	-13.9146
9xz	-11.7474(5)	-11.8873(57)	-13.9146
8y	unrealizable	-12.7447(37)	-13.9146

Figure 10: 3x3 lattice analysis.

* There is a wired thing in method c, need to discuss!

Method a is more stable than method b. Both of method a and b is far from the correct result, method c. As the time they take, method c takes several ms to finish; method a and b take several minutes and method a is 3 or 4 times faster than method b.

Now turn to larger lattice size, say, 7x7. It is wired in computation of method a that all these computations take exactly 1.5 days. By exactly, I means the error is at most 4 mins. The results are as below.

spins_be_flipped	/	49xy	49,51xy	47,49,51xy	47,49,51,53xy	47xz48xy49yz
vortices_distance	0	1	2	3	4	2
method_a_energy	-71.7927(30)	-69.779(34)	-67.673(37)	-65.574(40)	-63.598(42)	-70.053(39)
energy_diff	/	2.014	4.120	6.219	8.195	1.7397
method_c_energy	-77.1249	-76.8307	-76.8490	-76.8308	-76.8308	-76.8490
energy_diff	/	0.2942	0.2759	0.2941	0.2941	0.2759

Figure 11: Vortex vs. Distance analysis.

It is worth to evaluate energy of vertices by method c for more large systems.

vortices_distance	0	1	2	3	4	5	6
energy_for_7x7	-77.1249	-76.8307	-76.8490	-76.8308	-76.8308	-76.8490	
energy_diff	/	0.2942	0.2759	0.2941	0.2941	0.2759	
energy_for_25x25	-984.1179	-983.8513	-983.8205	-983.8551	-983.8108	-983.8209	-983.8310
energy_diff	/	0.2666	0.2974	0.2628	0.3071	0.2970	0.2869

Figure 12: Vortex vs. Distance analysis: Method C.

6 Conclusion

7 Acknowledgement

YMSC, Martin Duy Tat,

References

[1] A. Kitaev, *Anyons in an exactly solved model and beyond*, *Annals of Physics* **321** (2006) 2âĂŞ111.

[2] A. Kitaev, *Fault-tolerant quantum computation by anyons*, *Annals of Physics* **303** (2003) 2âĂŞ30.

[3] G. Jackeli and G. Khaliullin, *Mott insulators in the strong spin-orbit coupling limit: From heisenberg to a quantum compass and kitaev models*, *Physical Review Letters* **102** (2009) .

[4] K. S. Tikhonov, M. V. FeigelâĂŹman and A. Y. Kitaev, *Power-law spin correlations in a perturbed spin model on a honeycomb lattice*, *Physical Review Letters* **106** (2011) .

[5] D.-L. Deng, X. Li and S. Das Sarma, *Machine learning topological states*, *Physical Review B* **96** (2017) .

[6] G. Carleo, K. Choo, D. Hofmann, J. E. T. Smith, T. Westerhout, F. Alet et al., *Netket: A machine learning toolkit for many-body quantum systems*, *SoftwareX* (2019) 100311.

[7] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko and G. Carleo, *Neural-network quantum state tomography*, *Nature Physics* **14** (2018) 447ÃćÂĂÂŞ450.

[8] P. Jordan and E. P. Wigner, *About the Pauli exclusion principle*, *Z. Phys.* **47** (1928) 631.

[9] H.-D. Chen and Z. Nussinov, *Exact results of the kitaev model on a hexagonal lattice: spin states, string and brane correlators, and anyonic excitations*, *Journal of Physics A: Mathematical and Theoretical* **41** (2008) 075001.

[10] P. Schmoll and R. OrÃžs, *Kitaev honeycomb tensor networks: Exact unitary circuits and applications*, *Physical Review B* **95** (2017) .

[11] E. H. Lieb, *Flux phase of the half-filled band*, *Physical Review Letters* **73** (1994) 2158âĂŞ2161.

[12] N. Le Roux and Y. Bengio, *Representational power of restricted boltzmann machines and deep belief networks*, *Neural Computation* **20** (2008) 1631 [https://doi.org/10.1162/neco.2008.04-07-510].

[13] C. Nayak, S. H. Simon, A. Stern, M. Freedman and S. Das Sarma, *Non-abelian anyons and topological quantum computation*, *Reviews of Modern Physics* **80** (2008) 1083âĂŞ1159.

[14] G. Kour and R. Saabne, *Real-time segmentation of on-line handwritten arabic script*, in *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, pp. 417–422, IEEE, 2014.