Diagonalization of the one dimensional 1/2-spin isotropic Heisenberg spin chain via the Lanczos algorithm.

Reinaldo Magallanes Saunders*

December 2020

1 Introduction

The main objective of this assignment is to apply the Lanczos algorithm to the 1-dimensional Heisenberg model of ferromagnetism, with particular interest in the ground state. This will be done using a script in Python..

2 The Heisenberg spin chain

Spin chains consist of quantum spins located in a one dimensional lattice. One of the phenomenological models used to study magnetism is the so called Heisenberg spin chain or Heisenberg model. This model consists of a system of quantum spins $\hat{\mathbf{S}}^{(i)}$, located at lattice sites i. The Hamiltonian used to describe this system is

$$\hat{H} = -\sum_{(i,j)} J_x^{(ij)} \hat{S}_x^{(i)} \hat{S}_x^{(j)} + J_y^{(ij)} \hat{S}_y^{(i)} \hat{S}_y^{(j)} + J_z^{(ij)} \hat{S}_z^{(i)} \hat{S}_z^{(j)} \tag{1}$$

where the sum is over all pairs (i, j) of spins and $J_x^{(ij)}$, $J_y^{(ij)}$ and $J_z^{(ij)}$ are interaction factors.

The spin operators $\hat{S}_x^{(i)}$, $\hat{S}_y^{(i)}$ and $\hat{S}_z^{(i)}$ simply act on each site i, since they are of the form

$$\hat{S}_a^{(i)} = \hat{\mathbb{I}} \otimes \hat{\mathbb{I}} \otimes \cdots \otimes \hat{S}_a \otimes \cdots \otimes \hat{\mathbb{I}}, \quad a = x, y, z$$

where $\hat{S}_a$ is the i-th operator.

A simpler case is that of a single isotropic interaction factor. That is

$$J_x^{(ij)} = J_y^{(ij)} = J_z^{(ij)} = J \quad \forall (i, j)$$

.

*e-mail: rei.magallanes@gmail.com

If we also restrict to interaction between nearest neighbors and let N be the number of spins in the chain, we get:

$$\hat{H} = -J \sum_{\substack{(i,j) \\ j=i+1 \leq N}} \hat{\mathbf{S}}^{(i)} \cdot \hat{\mathbf{S}}^{(j)} \tag{2}$$

This Hamiltonian describes nearest neighbor spin-spin interaction. If $J > 0$ a ferromagnetic ground state, with all spins oriented along the same direction, will be favored. An anti-ferromagnetic ordering will be favored for $J < 0$.

There are two possible topologies for a one dimensional chain: open or closed. Correspondingly, there are two Heisenberg spin chain Hamiltonians describing nearest neighbor interaction:

$$\hat{H} = -J \sum_{i=1}^{N-1} \hat{\mathbf{S}}^{(i)} \cdot \hat{\mathbf{S}}^{(i+1)} \quad \text{(open)} \tag{3}$$

$$\hat{H} = -J \sum_{i=1}^{N-1} \hat{\mathbf{S}}^{(i)} \cdot \hat{\mathbf{S}}^{(i+1)} - J\hat{\mathbf{S}}^{(N)} \cdot \hat{\mathbf{S}}^{(1)} \quad \text{(closed)} \tag{4}$$

If we consider the usual half-spin raising and lowering operators

$$\hat{S_{\pm}} = \hat{S}_x \pm i\hat{S}_y$$

such that

$$\begin{aligned} \hat{S}_+ \left|\uparrow\right\rangle = 0, \quad \hat{S}_- \left|\uparrow\right\rangle = \hbar \left|\downarrow\right\rangle, \quad \hat{S}_z \left|\uparrow\right\rangle = \frac{\hbar}{2} \left|\uparrow\right\rangle \\ \hat{S}_+ \left|\downarrow\right\rangle = \hbar \left|\uparrow\right\rangle, \quad \hat{S}_- \left|\downarrow\right\rangle = 0, \quad \hat{S}_z \left|\downarrow\right\rangle = -\frac{\hbar}{2} \left|\downarrow\right\rangle \end{aligned} \tag{5}$$

we can write (2) as follows:

$$\hat{H} = -\frac{J}{2} \sum_{\substack{(i,j) \\ j=i+1 \leq N}} \hat{S}_+^{(i)} \hat{S}_-^{(j)} + \hat{S}_-^{(i)} \hat{S}_+^{(j)} + 2\hat{S}_z^{(i)} \hat{S}_z^{(j)} \tag{6}$$

The terms involving $\hat{S}_{\pm}$ are called hopping terms since they move a spin up or spin down to a neighboring site.

2.1 Hilbert Space

Problems with diagonalization start when the size of the matrix representation of the Hamiltonian is considered. The Hilbert space $\mathcal{H}$ of the system of N spins is formed by all possible linear combinations of products

$$\left|\upsilon\right\rangle = \left|\sigma_\upsilon\right\rangle_1 \otimes \left|\sigma_\upsilon\right\rangle_2 \otimes \cdots \otimes \left|\sigma_\upsilon\right\rangle_i \otimes \cdots \otimes \left|\sigma_\upsilon\right\rangle_N \tag{7}$$

where $|\sigma_\upsilon\rangle_i$ represents the two possible states for the i-th spin, either $|\uparrow\rangle_i$ or $|\downarrow\rangle_i$. Thus,

$$\dim(\mathcal{H}) = (2s+1)^N = 2^N$$

which translate to a matrix representation of the Hamiltonian of size $2^N \times 2^N$. So, for large systems it is pretty much impossible to diagonalize by brute force. For reference, for $N = 30$, $\dim(\mathcal{H}) \approx 10^9$.

One way to reduce the effective size of the Hamiltonian is by restricting the system to states of a given total magnetization. If we consider the operator

$$\hat{N} = \sum_i [\hat{S}_z^{(i)} + \hat{\mathbb{I}}]$$

which measures the total number of up spins, we can see that it commutes with the Hamiltonian. This means that we can restrict to subsets of a fixed number of spins up or down.

If $\mathcal{H}_n$ denotes the Hilbert subspace for n spins up, then:

$$\begin{aligned} \dim(\mathcal{H}) = 2^N = \dim(\mathcal{H}_1) + \dim(\mathcal{H}_2) + \cdots + \dim(\mathcal{H}_n) + \cdots + \dim(\mathcal{H}_N) \\ \dim(\mathcal{H}) = 2^N = N + \binom{N}{2} + \cdots + \binom{N}{n} + \cdots + 1 \end{aligned} \tag{8}$$

Thus, for a system of N spins with n spins up, $\dim(\mathcal{H}_n) = N!/(n!(N-n)!)$, which is considerably smaller. For reference, for $N = 30$, the maximum value for $\dim(\mathcal{H}_n)$ is $\dim(\mathcal{H}_{15}) \approx 1.5 \times 10^8$.

2.2 Sparseness of the Hamiltonian Matrix

For the Heisenberg spin chain, the number of non zero matrix elements of the Hamiltonian is much smaller than the total number of matrix elements. The matrix elements of the Hamiltonian will be given by

$$H_{\upsilon\eta} = \langle\upsilon|\hat{H}|\eta\rangle \tag{9}$$

where $|\upsilon\rangle$ and $|\eta\rangle$ are of the form (7) and $\hat{H}$ is given by (6).

From (5) we can see that the term $\hat{S}_z^{(i)}\hat{S}_z^{(j)}$ of the Hamiltonian will be diagonal, since both $|\uparrow\rangle$ and $|\downarrow\rangle$ are orthogonal eigenvectors of $\hat{S}_z$. So:

$$\langle\upsilon|\hat{S}_z^{(i)}\hat{S}_z^{(j)}|\eta\rangle = \pm\frac{\hbar^2}{4}\langle\upsilon|\eta\rangle \neq 0 \quad \text{if} \quad |\upsilon\rangle = |\eta\rangle \tag{10}$$

As for the hopping term, we see that:

$$|\sigma_\upsilon\rangle_k = |\sigma_\eta\rangle_k \, \forall k \neq i,j \implies \begin{cases} \langle\uparrow|_i \otimes \langle\downarrow|_j \, \hat{S}_+^{(i)}\hat{S}_-^{(j)} \, |\downarrow\rangle_i \otimes |\uparrow\rangle_j = \hbar^2 \neq 0 \\ \langle\downarrow|_i \otimes \langle\uparrow|_j \, \hat{S}_-^{(i)}\hat{S}_+^{(j)} \, |\uparrow\rangle_i \otimes |\downarrow\rangle_j = \hbar^2 \neq 0 \end{cases} \tag{11}$$

Thus, only these are non zero. Furthermore, since these matrix elements don't overlap, we can calculate them separately.

3 Lanczos Algorithm

In most cases we need only the lowest part of the eigenvalue spectrum of the Hamiltonian. If the matrix representation of the Hamiltonian is sparse, the Lanczos algorithm can be used since it yields only the lowest (or highest) eigenvalues and eigenvectors of a matrix with relative low computational power required.

The Lanczos algorithm works for Hermitian matrices, of which the Hamiltonian matrix for a quantum system is an example. Strictly speaking, the algorithm does not need access to the matrix explicitly, but rather only to a function that computes the product of said matrix by an arbitrary vector.

The starting point of the algorithm is a vector $|v_0\rangle = 0$ and another arbitrary vector $|v_1\rangle$ which is normalized. Then, generate the following vectors:

$$|w_{j+1}\rangle = \hat{H}\,|v_j\rangle - \alpha_j\,|v_j\rangle - \beta_j\,|v_{j-1}\rangle\,, \quad |v_{j+1}\rangle = \frac{1}{\beta_{j+1}}\,|w_{j+1}\rangle \tag{12}$$

where

$$\alpha_j = \langle v_j|\hat{H}|v_j\rangle\,, \quad \beta_{j+1} = \langle w_{j+1}|w_{j+1}\rangle^{\frac{1}{2}}$$

After p steps, with $p \leq \dim(\mathcal{H})$, the algorithm yields an orthonormal set of vectors $\{|v_j\rangle\}$. Furthermore, $\hat{H}$, when expressed with respect to the set $\{|v_j\rangle\}$, is a tridiagonal symmetric matrix with elements α_1, α_2, ..., α_p on the diagonal and elements β_2, β_3, ..., β_p on the off-diagonals.

The eigenvalues of the $p \times p$ tridiagonal matrix obtained after p iterations will converge to the eigenvalues of the original matrix. It can be shown that the lowest (and highest) eigenvalues converge the fastest. This is why the Lanczos algorithm is used for these parts of the spectrum. Since the diagonalization of a tridiagonal matrix is computationally quick, this algorithm gives a useful way to compute the lowest eigenvalues of a large matrix.

The main drawback of the Lanczos algorithm is that it is prone to numerical instability. This stems from the fact that, because of rounding errors, the orthogonality between the vectors $\{|v_j\rangle\}$ is lost rather quickly and, in some cases, linear dependence between vectors is present, both of which affect the eigenvalues obtained after diagonalization.

4 Algorithm and Implementation

The code will find the 3 lowest eigenvalues for the one dimensional Heisenberg spin chain. Both boundary conditions, either periodic or open, are implemented. Energies are in units of $|J|$ while spins are dimensionless.

The code treats the states of the system as binary representations of integers with a correspondence between spins (up or down) and bits (1 or 0). For instance, for a 3 site system, one possible state will be $|\uparrow\rangle_1 \otimes |\downarrow\rangle_2 \otimes |\uparrow\rangle_3$. This state is represented by the binary number 101. Thus, since $\dim(\mathcal{H}) = 2^N$, we can use the first $2^N - 1$ non negative integers $(0, 1, \ldots, 2^N - 1)$ to represent the possible states of the system. Due to *NumPy*'s limitations, this works only up to 64 spins. This can be prevented with libraries such as *mpmath*, used for real and complex floating-point arithmetic with arbitrary precision. Although, at that point problems with storage will be more important. This could be solved, as mentioned

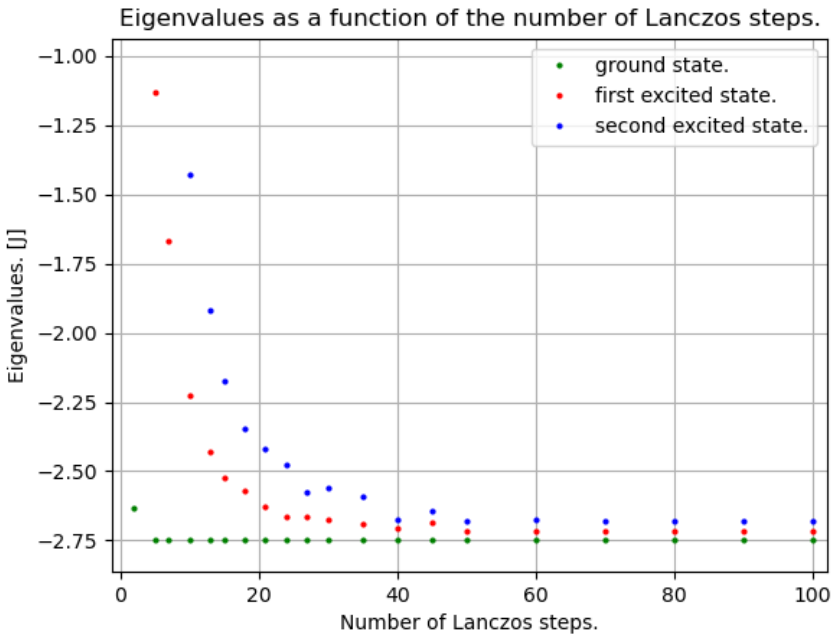

Figure 1: Eigenvalues vs number of steps in the Lanczos algorithm for the open ferromagnetic chain.

previously, by a function that calculates the product $\hat{H}\,|v_j\rangle$ in the Lanczos algorithm without storing the whole matrix.

5 Results and Discussion

First, to make sure the algorithm worked properly, we checked that the ground state energies obtained were the same as either theoretical values or references. For the ferromagnetic spin chain we compared against the theoretical values; $E_0 = -JN\hbar^2 s^2$ for the closed chain, while $E_0 = -J(N-1)\hbar^2 s^2$ for the open chain. For the anti-ferromagnetic spin chain we compared against values given by Thijssen[1] for the closed chain, namely $E_0 = -5.3873909$ for $N = 12$, and values given by Giannozzi[2] for the open chain, where values for E_0/N for $N = 2, 3, 4$ are provided.

Then, we examined the convergence of the Lanczos algorithm to the ground state energy and the energy of the first excited states. We took the open ferromagnetic case with $N = 12$ and $n = 6$ for the total number of spins and the number of up spins, respectively. Results can be seen in Figure 1. For the ground state the exact value was achieved, which is $E_0 = -(N-1)/4 = -2.75$. Convergence was achieved rather quickly. For 7 steps the value obtained was -2.74915844. For the excited states, convergence required significantly more steps. For example, for the first excited state, the exact value is $E_1 = E_0 + J\hbar^2(1 - \cos(\pi/N)) \approx -2.715926$, which was corroborated by conventional diagonalization, yielding -2.71592583. The earliest the obtained value was within 4 significant figures of this value was for 50 steps, which yielded -2.71583218.

It's worth noting that, because of the problems that the Lanczos algorithm has, when

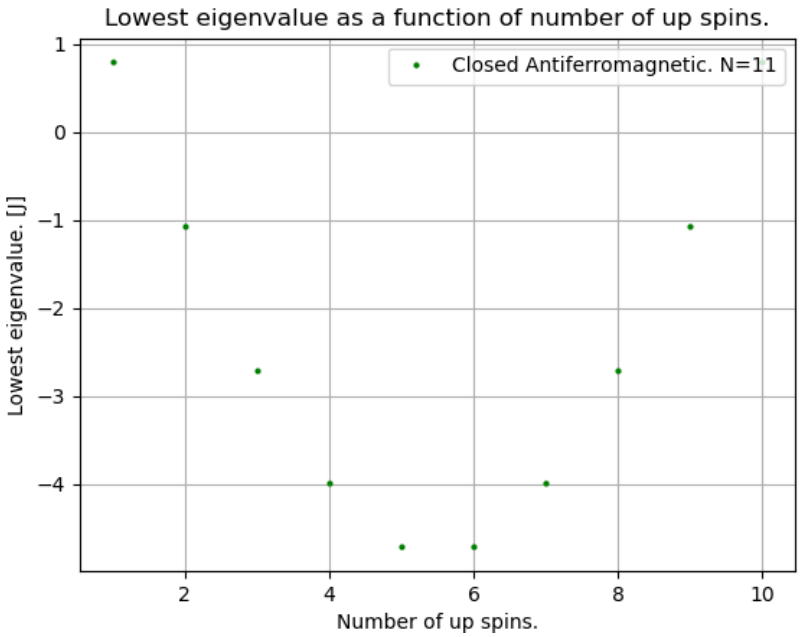

Figure 2: Lowest eigenvalue vs number of up spins for the closed anti-ferromagnetic chain ($N = 11$).

a large number of steps are performed, the excited states energy will tend towards lower energy states. This was first noticed for 135 steps. For 600 steps, all three values were the same, -2.75.

Next, for the closed ferromagnetic chain, it was verified that the ground state energy obtained does not depend upon the number n of up spins. Also, the case with $n = N-1$ that corresponds to the spin-wave solution was easily verified with conventional diagonalization, by corroborating that the coefficients of the eigenvectors were, in fact, $-1/\sqrt{N}$.

For the closed anti-ferromagnetic chain, we wanted to verify that the ground state has zero magnetization for even N and $1/2$ magnetization for odd N. This was done by taking values for N ranging from 3 to 12 and values for n from 1 to $N-1$. Results for $N = 11$ and $N = 12$ are shown in Figures 2 and 3, respectively. There is symmetry around $N/2$, so the lowest eigenvalues are obtained for $n = 5, 6$ for $N = 11$ and $n = 6$ for $N = 12$.

Finally, we wanted to see if the the gap $E_1 - E_0$ has a $1/N$ dependence, for the anti-ferromagnetic chain. To do this, values for E_1, E_0 and their difference were obtained for $N = 3, 4, \ldots, 12$. These, which were verified by conventional diagonalization, are shown in Figure 4. It's easy to see that, even though the gap seems to decrease with N, the cases for odd and even N should be treated separately. With this in mind, we fitted a curve of the form

$$E_1 - E_0 = A + \frac{B}{N}$$

to both cases using *SciPy*'s **curve_fit** function. We obtained $A = 0.038 \pm 0.004$, $B = 3.86 \pm 0.03$ for even N and $A = 0.53 \pm 0.07$, $B = 2.9 \pm 0.4$ for odd N. Both fits are shown in Figure 5. It can be seen that while for even N the fit is quite good, for odd N it appears that, while the gap does decrease with N, it does not follow a $1/N$ dependence.

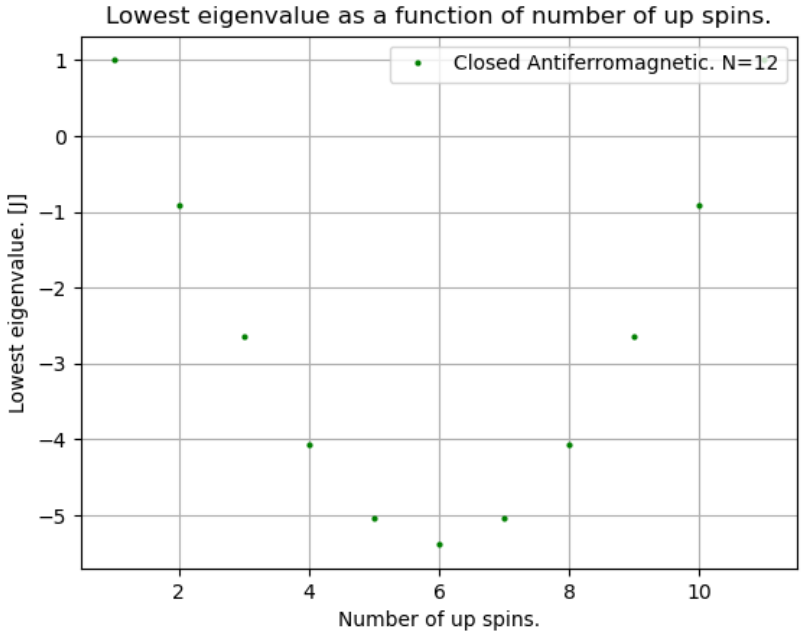

Figure 3: Lowest eigenvalue vs number of up spins for the closed anti-ferromagnetic chain ($N = 12$).

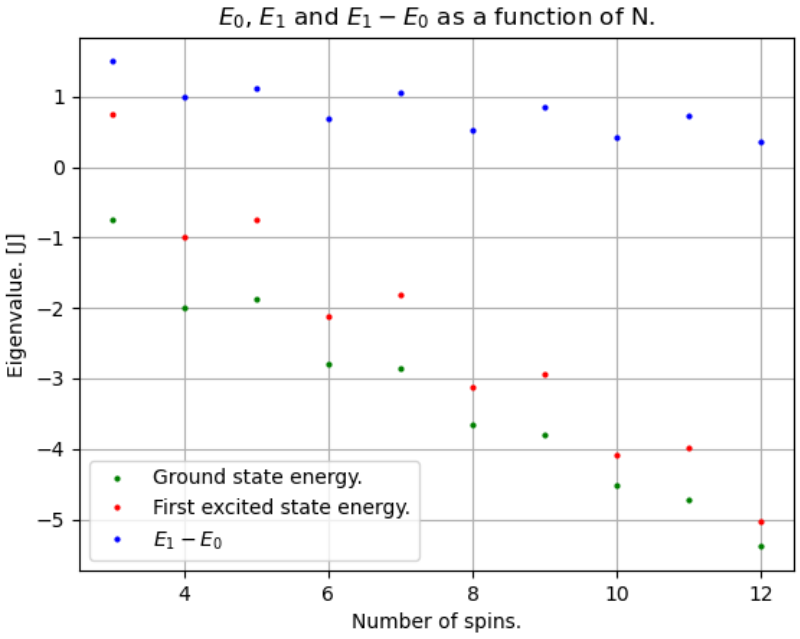

Figure 4: Ground state energy E_0, first excited state energy E_1, and $E_1 - E_0$ as a function of the number of spins for the closed anti-ferromagnetic chain.

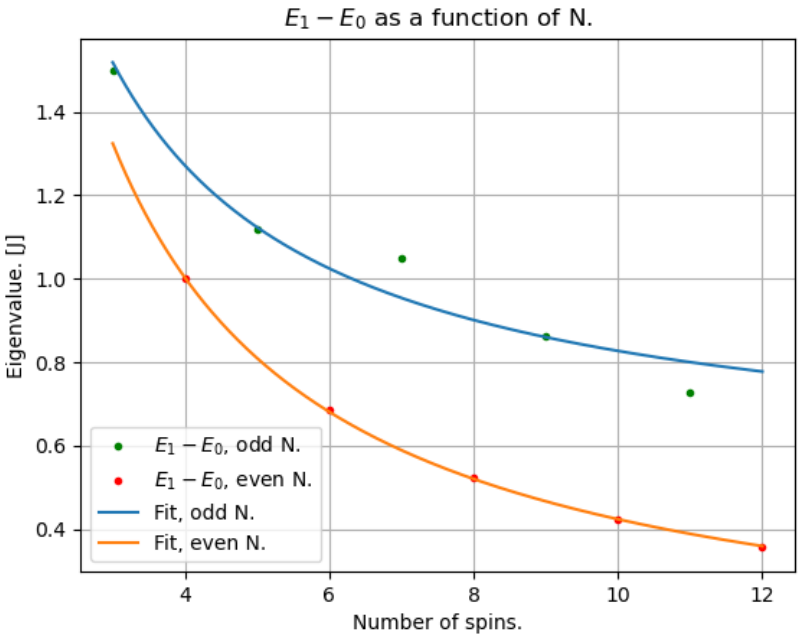

Figure 5: Fit for $E_1 - E_0$ for odd and even N for the anti-ferromagnetic chain.

References

[1] Thijssen, J. M. (2007) "*Computational Physics*". 2nd ed. Camdridge: Cambridge University Press.

[2] Giannozzi, P. (2019) "*Numerical Methods in Quantum Mechanics*". Lecture notes. University of Udine, Italy. Academic year 2018/2019.