Quantum information and Computing
Ex. 09

Andrea Nicolai - 1233407

December 14, 2020

Abstract

In this work we compute the Hamiltonian for a 1/2 spin N-particles 1-dim Ising model in a presence of a transverse external field and interaction between x-components of adjacent spins. We discuss the behavior of the first energy levels according to the value of λ, that describes the strength of interaction of z-components of spins.

Theory

The Hamiltonian for a N spin-1/2 particles on a 1-dim lattice is described by the Hamiltonian:

$$\hat{H} = \lambda \sum_{i}^{N} \sigma_z^i + \sum_{i}^{N-1} \sigma_x^i \sigma_x^{i+1}$$

where:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{1}$$

are the *Pauli* matrices and λ represents a linear chain of N interacting spins 1/2 in presence of an external field of intensity λ. This last term represents the effect of such field with the z component of the system, while the remaining describes an antiferromagnetic system.
Under Mean Field approximation, we assume that quantum correlations are neglected. The many-body wave function is constrained to be the product of N independent single body wave functions and can be rewritten as:

$$|\Psi_{MF}\rangle = \otimes_{i=1}^{N} |\psi^i\rangle = \otimes_{i=1}^{N} \sum_{\alpha_1=1}^{d} A_{\alpha_1} |\alpha_i\rangle$$

where, for 1/2 spin particles, we have that $\alpha_i = \uparrow, \downarrow$ indexed with $0, 1$ when dealing with qubits ($d = 2$).
System's energy is defined as:

$$E_{MF} = \langle \Psi_{MF} | \hat{H} | \Psi_{MF} \rangle = \sum_{i=1}^{N-1} (\langle \psi^1 | \sigma_x^i | \psi^1 \rangle)^2 + \lambda \sum_{i=1}^{N} (\langle \psi^1 | \sigma_z^i | \psi^1 \rangle) \tag{2}$$

Which is an extensive quantity, so we expect it to diverge in the thermodynamic limit. For this reason, we introduce the energy density $e = E/N$, that becomes:

$$e = -(\langle\psi^1|\sigma_x^i|\psi^1\rangle)^2 + \lambda(\langle\psi^1|\sigma_z^i|\psi^1\rangle) \tag{3}$$

One should note that the above expression is valid only in the thermodynamic limit, namely $N \to \infty$, but obviously we are limited by our hardware. Moreover, we will normalize wrt $N-1$ and not N in order to obtain a more consistent energy density: indeed in the thermodynamic limit this boundary effect would vanish and converge to a unique value. This is due to boundary effects. The energy density takes the value, in function of the external field λ:

$$\begin{cases} e = -1 - \lambda^2/4 & \lambda \in [-2, +2] \\ e = -|\lambda| & \lambda \notin [-2, +2] \end{cases} \tag{4}$$

Despite the latter is not exact, however, it well captures the physics of the system and correctly predicts the presence a phase transition even though not at the correct value. Indeed it denotes that for $\lambda = 2$ the second derivative of the energy density becomes a discontinuous function. However, from theory we would expect a quantum phase transition at $\lambda = 1$.

Code Development

Our Fortran program takes as input the maximum number of particles to be simulated, starting from a minimum of 2. There are two intervals over which we iterate: one is the number of the particles that is an *int* $\in [2, N_{max}]$, while the other one is *lambda* and, as said in theory section, is functional to compute the Hamiltonian of the system for different values of this parameters. We were asked to implement the code for $\lambda \in (0d0, 3d0)$: therefore we discretized the latter in 60 intervals. Hence resolution was $\Delta\lambda = 0.05$.
Once we have defined the two quantities over whose intervals we iterate, that we recall again being the number of particles *nn* and the λ, we started computing the Hamiltonian in two steps, according to which contribution they come from. The first contribution is the interaction with an external field, and λ describes the strength of it. It is a diagonal term, and it is given by the sum $\lambda \sum_i^N \sigma_z^i$. One should take into account that there are some implicit tensor product, that is the i-th term of this sum is:

$$\lambda \left(\bigotimes_{j=1}^{j-1} \right) \sigma_z^i \left(\bigotimes_{j=i+1}^{N} \right)$$

The code for this is the following, where *temp* is a matrix that takes into account the temporary resulting matrix:

```
!External field term
DO ii = 1, nn
```

```
!temp for the first iteration is the identity matrix
!unless we are considering the 1st particle
    IF (ii == 1) THEN
        temp = pauli_mat_z
    ELSE
        temp = INITIALIZE_IDENTITY_N(d_dim)
    END IF
    DO jj = 2, nn
        IF (jj == ii) THEN
            temp = TENSOR_PRODUCT(temp, pauli_mat_z)
        ELSE
            temp = TENSOR_PRODUCT(temp, identity_mat)
        END IF
    END DO
Hamiltonian%element = Hamiltonian%element + lambda*temp%element
DEALLOCATE(temp%element)
END DO
```

Where, as the name suggests, the function $TENSOR\_PRODUCT$ computes the tensor product between two tensors of rank 2. At the end of the day it was implemented as a *Kronecker product*.
Now it comes to compute the second term of the Hamiltonian, namely the one of interactions between two consecutive spins. One should note that each of the terms in the sum is a tensor product of two matrices σ_x, and is the one we stated above: $\sum_i^{N-1} \sigma_x^i \sigma_x^{i+1}$. Each of these $N-1$ terms looks like the following, considering the implicit tensor products:

$$\left(\bigotimes_{j=1}^{j-1}\right) \sigma_x^i \sigma_x^{i+1} \left(\bigotimes_{j=i+2}^{N}\right)$$

The following code compute the just mentioned interaction term.

```
!Interaction term of the Hamiltonian for two neighbours
DO ii = 1, nn - 1
    temp = INITIALIZE_IDENTITY_N(d_dim)
    DO jj = 2, nn
        IF ((jj == ii).OR.(jj == ii + 1)) THEN
            temp = TENSOR_PRODUCT(temp, pauli_mat_x)
        ELSE
            temp = TENSOR_PRODUCT(temp, identity_mat)
        END IF
    END DO
Hamiltonian%element = Hamiltonian%element + temp%element
DEALLOCATE(temp%element)
END DO
```

Once computed the Hamiltonian, we proceeded to its diagonalization and then save the first 4 eigenvalues and the actual value for λ to a ".dat" file, named according to the number of particles nn we are considering. As already said,

these procedure is iterated for whole $i \cdot \Delta\lambda,\ i\ = 0, ..., 60$ where $\Delta\lambda = 0.05$.
Finally we implemented some checkpoints for the parameter $N_{max} \geqslant 2$ and the hermitianity of $\hat{H}$ matrix before proceeding with its diagonalization.
A Python script, when executed, automatizes the compilation and the creation of all the plots shown in results section. The only parameter one may want to change is N_max, and has to be done inside the script.

Results

The program was run with the maximum number of particles it could simulate because of the RAM available on our laptop, namely $N_{max} = 11$. We recall that we chose to normalize wrt to $N - 1$, in order to take into account the boundary effects of the interaction term of the Hamiltonian, even though we know that the correct definition for the energy density has N in the denominator. This is due to boundary conditions: if we considered periodic boundary conditions, we should divide by N, having in this case the sum of N terms in the spin-spin interaction contribution.

As said, Hamiltonian has two contributions: interaction between x-components of two adjacent spins and the interaction of the z-component of every spin with the external field. The latter term has indeed symmetry $\mathcal{Z}^2$, this means that the lowest energy levels are the ones where all z-components of the spins are aligned either accordingly or oppositely with respect to the transverse field. Whereas the ground state for the former is the one where all x-components of spins alternate, however the first spin can assume either value 0 or 1 as the remaining ones keep alternating. This fact is shown in figures 2, 3, 4. For $\lambda = 0$ indeed the only surviving term is the x-component interaction and that has degeneracy 2 as just mentioned. However this symmetry is explicitly broken as $\lambda \neq 0$, and degeneracy is not present any more having we "switched on" the external field. As λ increases, there is a specific value where contributions given by different terms of the Hamiltonian start to overcome each other. Therefore we have a quantum phase transition and, from theory, we know that it should not be dependent on the size of the system ($\lambda = 1$). From that value on, eigenvalues do not seem to show degeneration anymore: this behavior can be observed specially when E_1 tends to move away from E_0 and cluster with the other eigenvalues E_2 and E_3. This does not happen when $\lambda = 2$ as predicted by the MF approach: indeed in $d = 1$ MF approach returns wrong results despite correctly predicts the presence of a phase transition. The other degeneration we observe is the one given by either one of the z-component breaks the alternation typical of the ground state, and is aligned accordingly to the external field. We expect these eigenvalues to have degeneracy N, since it can be achieved by flipping either one among N possible spins.
Last thing to notice is that, with particular referring to figure 1, the higher the number of the particle the closer the ground state to the MF energy prediction.

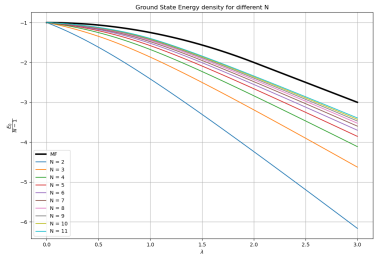

Fig. 1: Ground states energy for different number of particles.

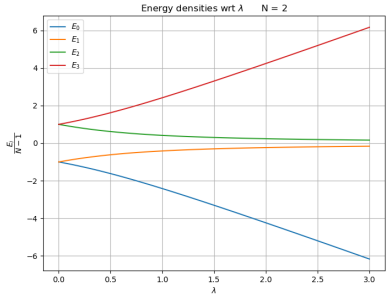

Fig. 2: First four eigenvalues for N = 2.

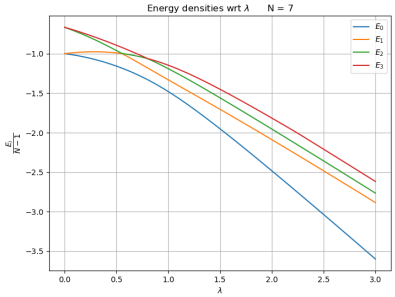

Fig. 3: First four eigenvalues for N = 7.

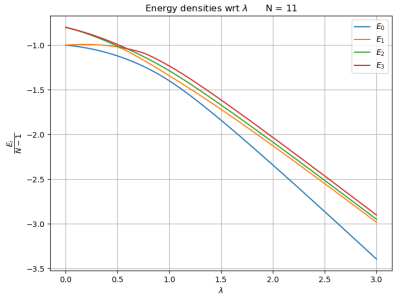

Fig. 4: First four eigenvalues for N = 11.

Self Evaluation

In this work we have studied the 1-dim Ising model for 1/2 spin particles in a transverse field, using the $|\Psi\rangle$ of the whole system. We have studied how its first eigenvalues according to the strength of interaction λ with the external field.
We wanted actually to go further and write the Hamiltonian in a a local basis in order to compute directly the element of $\hat{H}$, but since its diagonalization was not successful we abandoned this path, thus not including it in the final code attached to this report: using "qubit" all elements can generally be computed using logical operators (XOR, for instance). This must be more efficient: our algorithm shall seem slower because we compute many tensor products. There are for sure expressions explicit for matrix elements that would have improved the velocity of the program: the aim was to see how both approaches would have scaled wrt the size of the matrix. Since we were not able to finalize the second approach, we omitted this last part, despite it would be for sure interesting.