Chapter 11

Exact diagonalization of quantum spin models

Systems of interacting spins are used since many years to model magnetic phenomena. Their usefulness extends well beyond the field of magnetism, since many different physical phenomena can be mapped, exactly or approximately, onto spin systems. Many models exists and many techniques can be used to determine their properties under various conditions. In this chapter we will deal with the exact solution (i.e. finding the ground state) for the Heisenberg model, i.e. a quantum spin model in which spin centered at lattice sites interact via the exchange interaction. The hyper-simplified model we are going to study is sufficient to give an idea of the kind of problems one encounters when trying to solve many-body systems without resorting to mean-field approximations (i.e. reducing the many-body problem to that of a single spin under an effective field generated by the other spins). Moreover it allows to introduce two very important concepts in numerical analysis: iterative diagonalization and sparseness of a matrix.

11.1 The Heisenberg model

Let us consider a set of atoms in a crystal, each atom having a magnetic moment, typically due to localized, partially populated states such as 3\textit{d} states in transition metals and 4\textit{f} states in rare earths. The energy of the crystal may in general depend upon the orientation of the magnetic moments. In many cases these magnetic moments tend to spontaneously orient (at sufficiently low temperatures) along a given axis, in the same direction. This phenomenon is known as ferromagnetism. Other kinds of ordered structures are also known, and in particular antiferromagnetism: two or more sublattices of atoms are formed, having opposite magnetization. Well before the advent of quantum mechanics, it was realized that these phenomena could be quite well modeled by a system of interacting magnetic moments. The origin of the interaction was however mysterious, since direct dipole-dipole interaction is way too small to justify the

\footnote{but not for our model: it can be demonstrated that the magnetization vanishes at $T \neq 0$, for all 1-d models with short-range interactions only}
observed behavior. The microscopic origin of the interaction was later found in the antisymmetry of the wave functions and in the constraints it imposes on the electronic structure (this is why it is known as exchange interaction).

One of the phenomenological models used to study magnetism is the Heisenberg model. This consists of a system of quantum spins $S_i$, localized at lattice sites $i$, described by a spin Hamiltonian:

$$H = - \sum_{<ij>} (J_x(ij)S_x(i)S_x(j) + J_y(ij)S_y(i)S_y(j) + J_z(ij)S_z(i)S_z(j))$$

(11.1)

The sum runs over all pairs of spins.

In the following, we will restrict to the simpler case of a single isotropic interaction energy $J$ between nearest neighbors only:

$$H = -J \sum_{<ij>} S(i) \cdot S(j).$$

(11.2)

The restriction to nearest-neighbor interactions only makes physical sense, since in most physically relevant cases the exchange interaction is short-ranged. We will also restrict ourselves to the case $S = 1/2$.

### 11.2 Hilbert space in spin systems

The ground state of a spin system can be exactly found in principle, since the Hilbert space is finite: it is sufficient to diagonalize the Hamiltonian over a suitable basis set of finite dimension. The Hilbert space of spin systems is in fact formed by all possible linear combinations of products:

$$|\mu\rangle = |\sigma_\mu(1)\rangle \otimes |\sigma_\mu(2)\rangle \otimes \ldots \otimes |\sigma_\mu(N)\rangle$$

(11.3)

where $N$ is the number of spins and the $\sigma_\mu(i)$ labels the two possible spin states ($\sigma = -1/2$ or $\sigma = +1/2$) for the $i$-th spin. The Hilbert space has dimension $N_h = 2^N$ (or $N_h = (2S+1)^N$ for spin $S$), thus becoming quickly intractable for $N$ as small as a few tens (e.g. for $N = 30$, $N_h \sim 1$ billion). It is however possible to reduce the dimension of the Hilbert space by exploiting some symmetries of the system, or by restricting to states of given total magnetization. For a system of $N$ spins, $n$ up and $N - n$ down, it can be easily demonstrated that $N_h = N!/(n!/(N-n)!$. For 30 spins, this reduces the dimension of the Hilbert space to "only" 155 millions at most. The solution "reduces" (so to speak) to the diagonalization of the $N_h \times N_h$ Hamiltonian matrix $H_{\mu,\nu} = \langle \mu | H | \nu \rangle$, where $\mu$ and $\nu$ run on all possible $N_h$ states.

For a small number of spins, up to 12-13, the size of the problem may still tractable with today’s computers. For a larger number of spin, one has to resort to techniques exploiting the sparseness of the Hamiltonian matrix. The number of nonzero matrix elements is in fact much smaller than the total number of matrix elements. Let us re-write the spin Hamiltonian under the following form:

$$H = -\frac{J}{2} \sum_{<ij>} (S_+(i)S_-(j) + S_-(i)S_+(j) + 2S_z(i)S_z(j)).$$

(11.4)
The only nonzero matrix elements for the two first terms are between states $|\mu\rangle$ and $|\nu\rangle$ states such that $\sigma_\mu(k) = \sigma_\nu(k)$ for all $k \neq i, j$, while for $k = i, j$:

\begin{equation}
\langle \alpha(i) | \otimes \langle \beta(j) | S_+(i) S_-(j) | \beta(i) \rangle \otimes | \alpha(j) \rangle
\end{equation}

\begin{equation}
\langle \beta(i) | \otimes \langle \alpha(j) | S_-(i) S_+(j) | \alpha(i) \rangle \otimes | \beta(j) \rangle
\end{equation}

where $\alpha(i), \beta(i)$ mean $i$–th spin up and down, respectively. The term $S_z(i) S_z(j)$ is diagonal, i.e. nonzero only for $\mu = \nu$.

Sparseness, in conjunction with symmetry, can be used to reduce the Hamiltonian matrix into blocks of much smaller dimensions that can be diagonalized with a much reduced computational effort.

### 11.3 Iterative diagonalization

In addition to sparseness, there is another aspect that can be exploited to make the calculation more tractable. Typically one is interested in the ground state and in a few low-lying excited states, not in the entire spectrum. Calculating just a few eigenstates, however, is just marginally cheaper than calculating all of them, with conventional (LAPACK) diagonalization algorithms: an expensive tridiagonal (or equivalent) step, costing $O(N_h^3)$ floating-point operations, has to be performed anyway. It is possible to take advantage of the smallness of the number of desired eigenvalues, by using iterative diagonalization algorithms. Unlike conventional algorithms, they are based on successive refinement steps of a trial solution, until the required accuracy is reached. If an approximate solution is known, the convergence may be very quick. Iterative diagonalization algorithms typically use as basic ingredients $H\psi$, where $\psi$ is the trial solution. Such operations, in practice matrix-vector products, require $O(N_h^2)$ floating-point operations. Sparseness can however be exploited to speed up the calculation of $H\psi$ products. In some cases, the special structure of the matrix can also be exploited (this is the case for one-electron Hamiltonians in a plane-wave basis set). It is not just a problem of speed but of storage: even if we manage to store into memory vectors of length $N_h$, storage of a $N_h \times N_h$ matrix is impossible.

Among the many algorithms and variants, described in many thick books, a time-honored one that stands out for its simplicity is the Lanczos algorithm. Starting from $|v_0\rangle = 0$ and from some initial guess $|v_1\rangle$, normalized, we generate the following chain of vectors:

\begin{equation}
|w_{j+1}\rangle = 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,
\end{equation}

where

\begin{equation}
\alpha_j = \langle v_j | H | v_j \rangle, \quad \beta_{j+1} = (\langle w_{j+1} | w_{j+1} \rangle)^{1/2}.
\end{equation}

It can be shown that vectors $|v_j\rangle$ form an orthonormal set: $\langle v_i | v_j \rangle = \delta_{ij}$, and that in the basis of the $|v_j\rangle$ vectors, the Hamiltonian has a tridiagonal form,
with $\alpha_j$ elements on the diagonal, $\beta_j$ on the subdiagonal. After $n$ steps:

$$H_t = \begin{pmatrix}
\alpha_1 & \beta_2 & 0 & \ldots & 0 \\
\beta_2 & \alpha_2 & \beta_3 & 0 & \vdots \\
0 & \beta_3 & \alpha_3 & \ddots & 0 \\
\vdots & 0 & \ddots & \ddots & \beta_n \\
0 & \ldots & 0 & \beta_n & \alpha_n
\end{pmatrix}. \quad (11.9)$$

If $n = N_h$, this transformation becomes exact: $H_t = H$, and constitutes an alternative tridiagonalization algorithm. In practice, the Lanczos recursion tends to be unstable and may lead to loss of orthogonality between states. If however we limit to a few steps, we observe that the lowest eigenvalues, and especially the lowest one, of matrix $H_t$ converge very quickly to the corresponding ones of $H$. Since the diagonalization of a tridiagonal matrix is a very quick and easy operation, this procedure gives us a convenient numerical framework for finding a few low-lying states of large matrices. If moreover it is possible to exploit sparseness (or other properties of the matrix) to quickly calculate $H|v\rangle$ products \textit{without storing the entire matrix}, the advantage over conventional diagonalization becomes immense.

### 11.4 Code: heisenberg_exact

Code `heisenberg_exact.f90`\(^2\) (or `heisenberg_exact.c`\(^3\)) finds the ground state energy of the 1-dimensional Heisenberg model, using Periodic Boundary Conditions:

$$H = -J \sum_{i=1}^{N} \mathbf{S}(i) \cdot \mathbf{S}(i+1), \quad \mathbf{S}(N+1) = \mathbf{S}(1). \quad (11.10)$$

In the code, energies are in units of $|J|$, spins are adimensional. If $J > 0$ a ferromagnetic ground state, with all spins oriented along the same direction, will be favored, while the $J < 0$ case will favor an antiferromagnetic ordering. The sign of $J$ is set in the code (to change it, edit the code and recompile).

For the totally magnetized (ferromagnetic) case, the solution is trivial: there is just one state with all spins up (let us call it $|F\rangle$), yielding $E_0 = \langle F | H | F \rangle = -NJ/4$. Also the case with $N - 1$ spins up can be exactly solved. We have $N$ states with $N - 1$ spins up, that we label as $|n\rangle = S_{-}(n)|F\rangle$. Exploiting translational symmetry, one introduces Bloch-like states

$$|k\rangle = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} e^{ikn}|n\rangle, \quad k = 2\pi m/N, \quad m = 0, ..., N - 1. \quad (11.11)$$

It can then be shown that these are the eigenvectors of $H$ with eigenvalues $E(k) = E_0 + J(1 - \cos k)$. Careful readers will recognize \textit{spin waves} in this solution.

\(^2\)http://www.fisica.uniud.it/%7Egiannozz/Corsi/MQ/Software/F90/heisenberg_exact.f90
\(^3\)http://www.fisica.uniud.it/%7Egiannozz/Corsi/MQ/Software/C/heisenberg_exact.c
In the antiferromagnetic case, the ground state has $S_z = 0$ for even $S_z = 1/2$ for odd number of spins. In the limit of infinite chains, the ground-state energy is known: $E_0 = -NJ(\log 2 - 1/4)$, and the gap between $E_0$ and the first excited state $E_1$ decreases as $1/N$. A general exact solution (for other similar spin problems as well) can be found by means of the Bethe Ansatz, a highly nontrivial technique.

The code requires the number $N$ of spins and the number $nup$ of up spins, computes the dimension $nHil$ of the Hilbert space. It then proceeds to the labelling of the states, using a trick often employed for spin-1/2 systems: an integer index $k$, running from 1 to $2^N - 1$, contains in its $i$-th bit the information (0=down, 1=up) for the $i$-th spin. Of course this works only up to 32 spins, for default integers (or 64 spins for INTEGER(8)). The integer $k$ is stored into array states for the states in the required Hilbert space.

The Hamiltonian matrix is then filled (the code does not take advantage of sparseness) and the number of nonzero matrix elements counted. For the $S_+S_-$ and $S_-S_+$ terms in the Hamiltonian, only matrix elements as in 11.5 and 11.6, respectively, are calculated. We remark that the line

$$k = \text{states}(ii)+2**(j-1)-2**(i-1)$$

is a quick-and-dirty way to calculate the index for the state obtained by flipping down spin $i$ and flipping up spin $j$ in state states(ii).

We then proceed to the generation of the Lanczos chain. The number of chain steps (should not exceed $nHil$) is prompted for and read from terminal. The starting vector is filled with random numbers. Note the new BLAS routines dnorm2 and dgemv: the former calculates the module of a vector, the latter a matrix-vector product and is used to calculate $H|v\rangle$.

The Hamiltonian in tridiagonal form (contained in the two arrays d and e) is then diagonalized by the LAPACK routine dsterf, that actually finds only the eigenvalues. The lowest eigenvalues is then printed for increasing values of the dimension of the tridiagonal matrix, up to $nL$, so that the convergence of the Lanczos chain can be estimated. You can modify the code to print more eigenvalues.

As a final check, the matrix is diagonalized using the conventional algorithm (routine dspev). Note how much slower this final step is than the rest of the calculation! Once you are confident that the Lanczos chain works, you can speed up the calculations by commenting out the exact diagonalization step. The limiting factor will become the size of the Hamiltonian matrix.

11.4.1 Computer Laboratory

- Examine the convergence of the Lanczos procedure to the ground-state energy. Examine excited states as well: notice that they will invariably “fall down” towards lower-energy states after a sufficiently large number of Lanczos steps is performed, due to loss of orthogonality.

\footnote{A more elegant but hardly more transparent way would be to directly manipulate the corresponding bits.}
• For the antiferromagnetic case, verify that the ground state has zero magnetization (for even \(N\)) or magnetization 1/2 (for odd \(N\)). Plot the ground-state energy \(E_0\) and the first excited state \(E_1\) (where can you find it? why?) as a function of \(N\), try to verify if the gap \(E_1 - E_0\) has a \(1/N\) dependence. Note that all energies are at least doubly degenerate for odd \(N\), as a consequence of time-reversal symmetry and Kramer’s theorem.

• For the ferromagnetic case, verify that the ground state has all spins aligned. Note that the ground state will have the same energy no matter which total magnetization you choose! This is a consequence of the rotational invariance of the Heisenberg Hamiltonian. Verify that the case with \(N-1\) spins up corresponds to the spin-wave solution, Eq. (11.11). You will need to print all eigenvalues.

Possible code extensions:

• Modify the code in such a way that open boundary conditions (that is: the system is a finite chain) are used instead of periodic boundary conditions. You may find the following data useful to verify your results: \(E/N = -0.375\) for \(N = 2\), \(E/N = -1/3\) for \(N = 3\), \(E/N = -0.404\) per \(N = 4\), \(E/N \to -0.44325\) per \(N \to \infty\).

• Modify the code in such a way that the entire Hamiltonian matrix is no longer stored. There are two possible ways to do this:
  – Calculate the \(H\psi\) product “on the fly”, without ever storing the matrix;
  – Store only nonzero matrix elements, plus an index keeping track of their position.

Of course, you cannot any longer use dspev to diagonalize the Hamiltonian. Note that diagonalization packages for sparse matrices, such as ARPACK, exist.