Non-equilibrium dynamics

Exercise Sheet 12

When the Hilbert space of a quantum many-body system is too large to allow full diagonalization of the Hamiltonian $\hat{\mathcal{H}}$, one can use the Lanczos algorithm to iteratively time evolve any starting state with a controlled error. This algorithm is commonly used to obtain the low energy subspace of large Hamiltonians.

The algorithm goes as follows: Starting from an arbitrary normalized state $|v_0\rangle$, we recursively generate an orthonormal basis $|v_j\rangle$ by

$$|v_j'\rangle \equiv (\hat{\mathcal{H}} - \alpha_{j-1})\,|v_{j-1}\rangle - \beta_{j-1}\,|v_{j-2}\rangle\,, \tag{1}$$

$$|v_j\rangle \equiv \beta_j^{-1}\,|v_j'\rangle\,, \tag{2}$$

with $\beta_0 \equiv 0$ and

$$\alpha_j \equiv \langle v_j|\hat{\mathcal{H}}|v_j\rangle\,, \qquad \beta_j \equiv \left\|v_j'\right\|. \tag{3}$$

We stop when we have generated n vectors. If n is equal to the Hilbert space dimension, then the Hamiltonian matrix can be exactly decomposed in the orthonormal basis $|v_j\rangle$ as

$$\hat{\mathcal{H}} = VTV^\dagger, \tag{4}$$

where $T_{ij} = \langle v_i|\hat{\mathcal{H}}|v_j\rangle$ and $V_{ij} = \langle i|v_j\rangle$.

In practice, we only need a small amount of states ($n \sim 50$) for the ground state to be well approximated by the ground state of the truncated Hamiltonian T, which is easily obtainable.

We can use the Lanczos decomposition (4) when time-evolving a state from t to $t + \delta t$ as

$$|\Psi(t+\delta t)\rangle = e^{-i\hat{\mathcal{H}}\delta t}\,|\Psi(t)\rangle \approx V(t)e^{-iT(t)\delta t}\,|v_0(t)\rangle\,, \tag{5}$$

where we write V and T as time-dependent because we generated them by starting with $|v_0(t)\rangle = |\Psi(t)\rangle$. The exponential of T can be calculated after a numerical diagonalization of T.

1. Given that $\langle v_i|v_j\rangle = \delta_{ij}$ by construction, calculate T_{ij} and show that it is tridiagonal.

2. Let us implement the Lanczos algorithm applied to time-evolution as in Eq. (5). You may start with the provided script ex.py.

   (a) Write a function that will perform the Lanczos algorithm.

   Note: The most expensive operation of the algorithm is the matrix-vector multiplication $\hat{\mathcal{H}}\,|v_j\rangle$. Make sure you only perform this operation once per vector v_j.

   (b) Implement the time evolution by a step of δt for an input state $|\Psi(t)\rangle$.

3. Consider the transverse field Ising model on the torus of $N = L_x L_y$ spins:

$$\hat{\hat{\mathcal{H}}} = -\cos\frac{\pi x}{2}\sum_{\langle ij\rangle}\hat{\sigma}_i^z\hat{\sigma}_j^z - \sin\frac{\pi x}{2}\sum_j\hat{\sigma}_j^x, \tag{6}$$

with $0 \leq x \leq 1$. Write a function that will generate the Hamiltonian in the $\hat{\sigma}^z$ basis.

Suggestion: Make use of the fact that $\hat{\mathcal{H}}$ is a sparse matrix and store only the list of non-zero matrix elements. Generate the matrix using `scipy.sparse.csr_matrix`.

4. Time evolve product states of your choice ($\hat{\sigma}^z$ configurations, for example) and calculate the average number of up and down spins ($n_\uparrow$ and $n_\downarrow$) at each time step. Plot these quantities over time and compare extreme energy to mid-spectrum states.

   Suggestion: Take $t_f = 5$, $\delta t = 0.05$, $N \leq 16$, $n = 20$, and $x \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$.

5. (Optional) Repeat the above analysis for the non-integrable 1D Ising model:

$$\hat{\hat{\mathcal{H}}} = -\sum_j \hat{\sigma}_j^z \hat{\sigma}_{j+1}^z + h_x \hat{\sigma}_j^x + h_z \hat{\sigma}_j^z. \tag{7}$$