Projects marked with star are a bit more elaborate. Projects are assigned on a first come first serve basis.

Project 1: N-state Potts model (Monte Carlo)

Implement the Metropolis algorithm for the N-state Potts model in 2D and observe/discuss the behaviour of the phase transition for different N. The N-state Potts model is a generalization of the Ising model, where each s_i takes N possible values, and the Hamiltonian is given by $H = -J \sum_{\langle i,j \rangle} \delta_{s_i,s_j}$, where $\langle i,j \rangle$ denotes nearest neighbours and δ_{s_i,s_j} the Kronecker-delta.

Project 2: Criticality of the (classical) 3D Ising model (Monte Carlo)

Generalize the Swendsen-Wang algorithm to the (classical) Ising model on a 3D cubic lattice and perform a finite size scaling analysis to obtain the critical exponents.

Project 3*: The worm algorithm for the 6-vertex model (Monte Carlo)

Implement the worm algorithm for the 6-vertex model (`https://en.wikipedia.org/wiki/Ice-type_model`) at $T = \infty$ (where all allowed configurations are equally likely). Calculate the exponent a of the correlation function $\langle s_0 s_r \rangle \propto \frac{1}{r^a}$.

Project 4*: Ground state properties of the 2D (transverse field) Ising model with Conservation laws (Exact Diagonalization)

Generalize (the code of) exercise 5 to the transverse field Ising model in 2 spatial dimensions, use both k_x and k_y as quantum numbers. Find the ground state (and possibly a few excited states) to locate the quantum phase transition.

Project 5: Krylov time evolution in a random Heisenberg chain (Exact Diagonalization)

The Lanczos algorithm can be used to find the ground state of H projected into the Krylov subspace and transformed it back to the full Hilbert space to find a good approximation of the ground state in the full Hilbert space. In a similar fashion, one can do a time evolution in the Krylov subspace to get an approximation of the time evolution in the full Hilbert space (`https://www.jstor.org/stable/2158085`). Building up on a provided `lanczos.py` implementing the Lanczos algorithm, use this method to perform the time evolution of a product state (e.g. $|\uparrow\downarrow\uparrow\downarrow\ldots\rangle$) for the Heisenberg model with a random field, $H = J\sum_i \vec{S}_i \cdot \vec{S}_{i+1} - \sum_i h_i S_i^z$, where the values of the field $h_i \in [-W, W]$ are chosen from a uniform random distribution with a "disorder strength" W. Plot the growth of the entanglement entropy $S(t)$ for small $W = 0.5J$ and for large $W = 5J$.

Project 6*: Continued fractions for dynamical properties (Exact Diagonalization)

As discussed in the lecture notes, one can evaluate continued fractions based on the tridiagonal matrix returned by the Lanczos algorithm to get quantities of the form

$$I(\hat{O}, z) = -\frac{1}{\pi}\mathrm{Im}\,\underbrace{\langle\psi_0|\hat{O}^\dagger \frac{1}{z-H}\hat{O}|\psi_0\rangle}_{\equiv x_0} \qquad \text{with } z = \omega + E_0 + \mathrm{i}\epsilon, \tag{1}$$

where $|\psi_0\rangle$ is a ground state. Build up on a provided `lanczos.py` implementing the Lanczos algorithm (c.f. project above), to evaluate $I(S_0^+, \omega)$, where $S_j^+ = \frac{1}{2}(\sigma_j^x + \mathrm{i}\sigma_j^y)$ for the ground states of the transverse field Ising model and the Heisenberg model (with periodic boundary conditions). Moreover, evaluate and (2D color-) plot $I(S_k^+, \omega)$ for the different k values of $S_k^+ = \frac{1}{\sqrt{L}}\sum_{j=0}^{L-1} e^{\mathrm{i}jk} S_j^+$.

Project 7: Construction of the reduced density matrix (MPS)

Use DMRG to calculate the ground state $|\psi\rangle$ of the transverse field Ising model for a large system ($L \approx 100$) as an MPS. Find a method to construct the reduced density matrix of n sites ($n \lesssim 10$) in the center. Measure the entanglement entropy of this subsystem of n sites for the different n.

Project 8*: Purification (MPS)

Purification (see e.g. the review by Schollwoeck) is a method to represent mixed states with MPS, which can be used to calculate thermodynamic properties from $\rho = e^{-\beta H}$. Calculate the energy, specific heat and magnetization as a function of temperature for the transverse field Ising model.

Project 9: Dynamical correlation functions (MPS)

Use the TEBD and DMRG to calculate correlation functions of the form $\langle\psi_0|\, e^{\mathrm{i}Ht} S_i^+ e^{-\mathrm{i}Ht} S_j^- \,|\psi_0\rangle$, where $|\psi_0\rangle$ is the ground state of the transverse field Ising model. Perform a fourier transformation in space and time and compare your results to the results of exercise sheets 5/6.

Project 10*: DMRG for fermions (MPS)

Make use of the Jordan-Wigner transformation to write the Fermi-Hubbard Hamiltonain $H = -t\sum_{i,\sigma}(c_{i,\sigma}^\dagger c_{i+1,\sigma} + h.c.) - U\sum_i n_{i,\uparrow} n_{i,\downarrow}$ with fermionic creation operatos $c_{i,\sigma}^\dagger$ and

$n_i = c_i^\dagger c_{i,\sigma}$ as an MPO. Use DMRG to find the ground state at half filling (i.e. for $N = L$ particles, where L is the number of sites, $i = 0, \dots L-1$) and calculate correlation functions $\langle c_{i\sigma}^\dagger c_{j\tilde{\sigma}} \rangle$ for different interaction strengths U. Some explanations: `https://tenpy.github.io/intro_JordanWigner.html`.

Project 11: Phase diagram of the 1D Bose-Hubbard model (MPS)

Use DMRG to find the ground states of the interacting 1D Bose-Hubbard model $H = -t\sum_i (a_i^\dagger a_{i+1} + h.c.) + \frac{U}{2}\sum_i n_i(n_i - 1) - \mu \sum_i n_i$ for different choices of the parameters t, U, μ. Note that you need to cut off the maximum number of bosons on each site to $n_c = 1, 2, 3, \dots$. Calculate the correlation function of superfluid order $\langle a_i^\dagger a_{i+r} \rangle$ and the density $\langle n_i \rangle$ and use them to determine the ground state phase diagram. Compare the influence of different cutoffs n_c on the result.

Project 12: Heisenberg model with short and exponential decaying interactions (MPS)

There is a natural way to write down matrix product operators for interactions which decay exponentially in distance, here we consider $H = \sum_i \sum_{j>i} J e^{-\frac{|i-j|}{\xi}} \vec{S}_i \cdot \vec{S}_j$. Use DMRG to find the ground state. Find out which influence ξ has on the correlations $\langle S_i^z S_j^z \rangle$.

Project 13*: Time-evolution using the time-dependent variational principle (MPS)

An alternative for TEBD as an algorithm for real-time evolution is based on the time-dependent variational principle (TDVP). A description of the implementation of the algorithm can be found in `https://arxiv.org/pdf/1408.5056.pdf`. Implement this algorithm and apply it on the transverse field Ising model, i.e. make a quench starting from the initial state $|\uparrow\uparrow \dots \uparrow\uparrow\rangle$. What are the advantages of using TDVP? Compare your results to what you get by using the TEBD implementation from the tutorials.

Project 14*: Machine Learning of Many Body Localization (Exact diagonalization + Machine Learning)

Use exact diagonalization to obtain all eigenstates of the the Heisenberg model with a random field, $H = J\sum_i \vec{S}_i \cdot \vec{S}_{i+1} - \sum_i h_i S_i^z$, where the values of the field $h_i \in [-W, W]$ are chosen from a uniform random distribution with a "disorder strength" W (Use moderate system sizes $L = 10, 12$). The exciting property of this model is that it is believed to undergo a phase transition from an extended phase (small W) to a localized phase (large W). We will use ML to detect this transition: Pick a number of eigenstates that are near energy $E = 0$ and obtain the reduced density matrices ρ^A, where A is a region of n consecutive spins (a few hundred to thousands eigenstates for different disorder realizations). Now use the density matrices for $W = 0.5J$ and $W = 8.0J$ to train a neural network (just interpret the entries of ρ^A as an image with $2^n \times 2^n$ pixel). Then use this network and study the output of the neural network for different W. How does the results depend on system size L and block size n? At which W_c do you expect the transition to occur?

Project 15: PEPS with simplified update on a square lattice (PEPS)

Generalize the code from exercise 9.2 (PEPS with simplified update) to (approximately) get the ground state of the transvere field Ising model on the square lattice. In addition to the on-site expectation values, evaluate (the decay of) correlation functions along the direction of the boundary MPS. Use the decay of correlation functions to locate the transition to the symmetry broken state.