# Tensor Networks Assignment 2025/26

February 5, 2026

## 1 Context: Time evolution of quantum many-body systems using MPS

We consider a simple quantum quench and study the real-time evolution of the transverse-field Ising model, a one-dimensional spin-1/2 chain with open boundary conditions. The Hamiltonian is

$$H_{\text{Ising}}(g) = \sum_{i=1}^{N-1} X_i X_{i+1} + g \sum_{i=1}^{N} Z_i, \tag{1}$$

where $X_i$ and $Z_i$ denote the Pauli matrices $\sigma_x$ and $\sigma_z$ acting on site $i$.

Starting from an initial state $|\psi_0\rangle$, we are interested in computing the expectation value of a local operator,

$$\langle O(t)\rangle = \langle \psi(t)|O|\psi(t)\rangle\,, \qquad |\psi(t)\rangle = e^{-iHt}\,|\psi_0\rangle\,. \tag{2}$$

As discussed in class, one way to perform time evolution using matrix product states (MPS) is to Trotterize the evolution operator $U(t) = e^{-iHt}$ into small time steps $U(\delta t)$, and represent each step as a matrix product operator (MPO). The two-body gates appearing in the Trotter decomposition can either be factorized using singular value decompositions, or constructed analytically (e.g. for $\exp(-iX_iX_{i+1}\delta t)$). One-body terms can be included using either first-order or (preferably) second-order symmetric Trotterization, which has the same computational cost but improved error scaling [1, 2].

The resulting MPOs are then applied sequentially to the MPS representing the state, truncating the bond dimension at each step to a maximum value. As extensively discussed in class, this truncation is performed by retaining the dominant eigenvalues of reduced density matrices across each bipartition, in close analogy with the DMRG algorithm. Standard references include [3, 4], as well as those in your Perusall. Once the evolved state $|\psi(t)\rangle$ is obtained, expectation values such as $\langle\psi(t)|O|\psi(t)\rangle$ can be computed straightforwardly.

The tensor network associated with this computation is shown schematically in Fig. 1. While the most common approach is to contract this network along the time direction (as in TEBD), other contraction strategies are possible. For instance, one may evolve the operator in the Heisenberg picture and evaluate $\langle\psi_0|O(t)|\psi_0\rangle$.

A third possibility, which will be central to this assignment, is to contract the spacetime tensor network along the *spatial* direction. In this approach, one groups tensors column by column, interpreting each column as a transfer matrix in MPO form, which we denote by $E$. Each transfer matrix contains tensors coming both from $U(\delta t)$ and its conjugate $U^\dagger(\delta t)$, see Fig. 1.

By introducing left and right boundary MPS, $\langle L_0|$ and $|R_0\rangle$, one can iteratively apply the transfer matrix $E$ to these boundaries, contracting from the edges toward the center of the system and truncating after each step. This will give us our left and right MPS $\langle L|$ and $|R\rangle$. When we reach the column containing the local operator, denoted as $E_{\mathrm{op}}$, the expectation value is finally obtained as

$$\langle O(t)\rangle = \langle L|E_{\mathrm{op}}|R\rangle \,. \tag{3}$$

# 2 The assignment

The goal of this assignment is to compute the same physical observable using two different tensor-network contraction strategies: (i) standard real-time evolution with MPS (TEBD), and (ii) spatial contraction of the corresponding tensor network.

## 2.1 General remarks

- Read the full assignment before starting, so you can plan ahead and reuse elements from earlier parts.
- You may use your preferred tensor-network framework: either low-level implementations with NumPy arrays and `ncon` / `np.einsum`, or higher-level libraries providing basic MPS operations such as `TeNPy` or `ITensors`.
- Work in a Jupyter notebook and export it as a PDF at the end, clearly showing the steps leading to the final results.
- It is unrealistic to expect that no AI tools will be used. Assistance with syntax or debugging is acceptable; vibe-coding without understanding what is going on can be easily spotted and will be negatively evaluated.

## 2.2 Setup

We consider a system of $N = 50$ spins with open boundary conditions, initialized in the product state

$$|\psi_0\rangle = \bigotimes_{i=1}^{N} |\uparrow\rangle_i \,, \tag{4}$$

where $|\uparrow\rangle$ is the $+1$ eigenstate of $Z$.

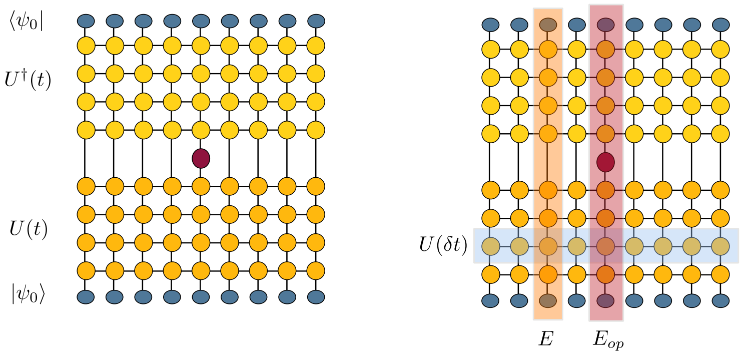

Figure 1: The 2D tensor network associated with the time evolution for an expected value of a local operator in the middle of a one-dimensional spin chain. If we contract in the temporal direction, our basic objects will be the initial states and the rows of the TN. On the other hand, for a contraction in the spatial direction we will work with

The system is quenched with the Ising Hamiltonian with $g = 0.7$ and evolved up to a final time $T = 4$, using a Trotter time step $\delta t = 0.1$ (or $\delta t = 0.05$ for increased accuracy). The maximum bond dimension is fixed to $\chi = 200$ throughout.

Unless otherwise stated, local observables are measured at the central site $i = N/2$.

## 2.3 Part I (3 pts): Expectation value with TEBD

Compute the expectation value of the local operator $Z_{i=N/2}(t)$ up to time $T = 4$ using TEBD.

There are (at least) two ways to do it, both acceptable:

- Construct the Trotterized MPO building blocks for $U(t)$ explicitly, apply them to the MPS, and truncate after each step.
- Use and adapt the TEBD example provided by TeNPy [5].

Plot $\langle Z_{N/2}(t)\rangle$ as a function of time.

## 2.4 Part II (6 pts): Expectation value via transverse contraction

Using the same quench parameters, compute the same expectation value by contracting the spacetime tensor network along the spatial direction.

1. Identify columns of the tensor network as transfer matrices $E$ in MPO form. You may reuse the tensors defined in Part I, keeping in mind that the roles of physical and virtual legs are now exchanged.

2. Initialize left and right boundary states $\langle L_0|$ and $|R_0\rangle$ as MPS. Iteratively apply the transfer matrix $E$ from both sides, truncating up to bond dimension $\chi = 200$ if necessary, until reaching the column containing the local operator. This defines the effective states $\langle L|$ and $|R\rangle$.

   **Technical remarks:**

   - If your MPO tensors are left-right symmetric (this can be done, as we have discussed in class), you may work with a single boundary MPS, since the other is simply its transpose (note: transpose, not the Hermitian conjugate!). Otherwise, just take care to apply the MPO in the correct direction and compute both $|R\rangle$ and $\langle L|$.
   - Since the local operator lies deep inside the light cone and here the Lieb–Robinson velocity satisfies $v < 5$, boundary effects are irrelevant for the times considered, so we could treat the system as if it was infinite. Our solution for $\langle L|$ and $|R\rangle$ could then be seen as the result of a "power method", as we have seen in class: we just apply the transfer matrix $E$ many times until convergence is reached - in this context, do you expect the actual choice of the initial boundary MPS to matter ? Discuss what you think and why, and check numerically if you're right (you can try eg. comparing with random MPS at the boundaries).
   - Unlike $U(\delta t)$, the transfer matrix $E$ is not unitary. Repeated application of it may thus cause the norm of the MPS to grow or decay exponentially. To have a stable algorithm, you might need to normalize the MPS after each step; any remaining normalization factors can be handled with the following prescription for computing the expectation value.

3. Compute the expectation value as

   $$\langle O(t)\rangle = \frac{\langle L|E_{\text{op}}|R\rangle}{\langle L|E|R\rangle}, \tag{5}$$

   where $E_{\text{op}}$ denotes the column containing the local operator and $E$ the corresponding column without it. Verify that the result agrees with Part I.

## 2.5 Part III (1+1 pts): Entanglement of transverse MPS

Plot the von Neumann entanglement entropy across the bonds of the left and right MPS obtained from the spatial contraction.

For one additional point, repeat the same TN contraction for different final times (e.g. $T = 0.1$ up to $T = 4$), and plot the maximum entanglement entropy of $|R\rangle$ as a function of time. How does it scale?

# References

[1] V. Murg, J.I. Cirac, B. Pirvu, F. Verstraete, arXiv:0804.3976

[2] Sebastian Paeckel et al, arXiv:1901.05824

[3] https://www.tensornetwork.org

[4] U. Schollwoeck, arXiv:1008.3477

[5] `https://tenpy.readthedocs.io/en/latest/notebooks/00_tebd.html`