Mini-project: TEBD for Rydberg Atom Arrays

Submitted to Marin Bukov and Markus Schmitt by Moksh Bhateja

December 11, 2024

GitHub Repository

The goal of this project is to implement the Time-Evolving Block Decimation (TEBD) algorithm to study the quench dynamics of a quasi-1D chain of Rydberg atoms. The ultimate goal is to reproduce Fig. 4 of arXiv:2405.21019.

1 Introduction

In their recent work Lukin et al. developed and tested a novel protocol "sweep-quench-sweep" (SQS) to aid the previously existing quantum adiabatic algorithm (QAA) (which uses a simple linear sweep protocol). The QAA, which transforms an initial easily prepared ground state to a target state by slowly varying some parameters of the Hamiltonian, fails to tackle systems with superexponentially small gap between the ground state and the first excited state due to a diabatic transition to an excited state at a small gap along the path. **The SQS protocol mitigates this problem by performing a quench prior to crossing the small gap which transfers a macroscopic state fraction to the first excited state that diabatically connects to the ground state during the subsequent linear sweep.**

In their work, Lukin et al. specifically focuses on Ising chains with long-ranged interactions and constrained dynamics arranged in particular quasi-1D and 2D geometries using Rydberg atoms. The many-body Hamiltonian for a Rydberg quantum simulator is given by

$$\frac{H_{\text{Ryd}}}{\hbar} = \frac{\Omega}{2}\sum_i \sigma_i^x - \Delta\sum_i n_i + V_0\sum_{i,j}\frac{n_i n_j}{|\mathbf{r}_i - \mathbf{r}_j|^6} \tag{1.1}$$

For the given quasi 1D lattice fig. 1:

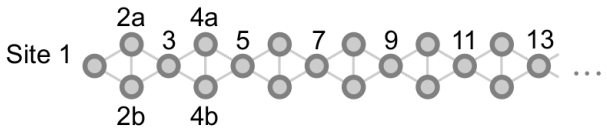


Figure 1: Quasi 1D lattice

The initial state of the system is given as: $|g\rangle_1 |gg\rangle_2 |g\rangle_3 |gg\rangle_4 \cdots |gg\rangle_{L-1} |g\rangle_L$, where L is the system size.

The target state is of the form:

$$|\text{MIS}\rangle \equiv |r\rangle_1 |gg\rangle_2 \left(\prod_{i \text{ odd}} |r\rangle_i \right) (|gg\rangle_i) |gg\rangle_{L-1} |r\rangle_L \tag{1.2}$$

$$|\text{Zigzag}\rangle \equiv |r\rangle_1 |gg\rangle_2 \left(\prod_{i \text{ odd}} |g\rangle_i \right) \left(\prod_{i \text{ even}} \frac{|rg\rangle_i + |gr\rangle_i}{\sqrt{2}} \right) |gg\rangle_{L-1} |r\rangle_L \tag{1.3}$$

where $|g\rangle$, $|r\rangle$ are the ground state and a highly excited Rydberg state respectively.

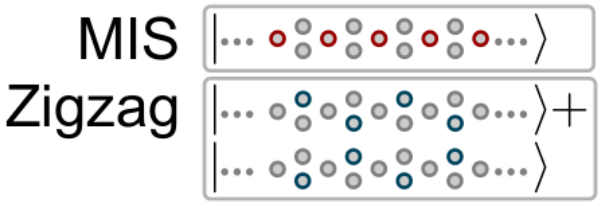


Figure 2: MIS and Zigzag state visualization

2 TEBD

The custom TEBD code was written in python using the *ncon* function [14] [3]. Left canonical form was chosen for the Matrix-product-state (MPS) representation. The unfolding of the quasi lattice was as follows: fig. 3

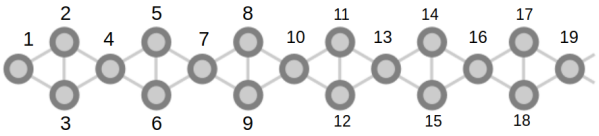


Figure 3: Unfolding of the Quasi 1D lattice

The code was bench-marked and developed in the following order. For each test relative error of 10^{-6} was achieved.

1. The code was developed for NN interactions and was tested for the Quantum XX Hamiltonian to get the ground state energy using imaginary time evolution.

2. NNN interactions were taken into account by keeping the 3 body Hamiltonian terms together [13] [16]. That is, the Hamiltonian block were created with 6 legs, each having $d = 2$ dimensions. This was also used to simulate the PXP Hamiltonian ([7] [10]) fig. 4, but the quasi structure of the lattice was ignored during this simulations. Including the quasi structure for the 3 body Hamiltonian was tedious and was hence ignored.

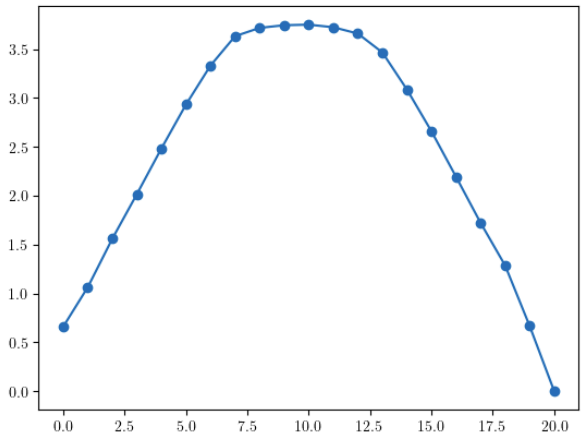


Figure 4: Entanglement entropy vs bond index for the PXP Hamiltonian.

3. NNN interactions were taken into account by keeping only 2 body Hamiltonian terms together. This was achieved by having SWAP gates between sites. The SWAP operation just amounts to have the legs of the effective site gate being swapped [18] [20]. The code was used to simulate the Heisenberg J1-J2 model for random values of J1-J2 to get the ground state energy using imaginary time evolution. Here the (Suzuki-)Trotter decomposition was crucial and was considered as [12]:

$$e^{(A+B+\cdots)h} = e^{Ah}e^{Bh}\cdots e^{\mathcal{O}(h^2)}$$

That is, the sweep is performed as $P^T_{\Delta t} = \hat{U}_{1,2}\hat{U}_{2,3}\cdots\hat{U}_{N-1,N}$. To have 2nd order decomposition we have $P^T_{\Delta t/2}P_{\Delta t/2}$. We can also have higher order decompositions [17], for example the 3rd order is given as $P^T_{\Delta t/9}P_{\Delta t/9}P_{\Delta t/9}P_{\Delta t/9}P^T_{\Delta t/9}P_{-\Delta t/18}P_{\Delta t/9}P_{\Delta t/9}P_{\Delta t/9}$. It is important to note that the orthogonality of the left canonical form is maintained by applying the gates in reverse order, or otherwise a series of identity gates must be applied between sweeps ($\not\hat{U}_{1,2}\not\hat{U}_{2,3}\cdots\not\hat{U}_{N-1,N}$). For 2nd order (which we use), this is not required.

4. Finally the code was modified to have long-range interactions (as we have for the quasi 1D lattice, because of the unfolding). This amounts to have multiple SWAP gates and modifying the $P^T_{\Delta t}$ as follows

$$\hat{P}^T_{\Delta t'} = \begin{pmatrix} \hat{U}_{1,2} & \hat{U}_{1,3} & \hat{U}_{1,4} & \cdots & \hat{U}_{1,1+l} \\ \hat{U}_{2,3} & & \hat{U}_{2,4} & & \hat{U}_{2,2+l} \\ & \ddots & & & \vdots \\ \hat{U}_{m,m+1} & & \hat{U}_{m,m+2} & & \hat{U}_{m,N} \\ & & \hat{U}_{m+1,m+2} & & \hat{U}_{m+1,N} \\ & & & \ddots & \vdots \\ & & & & \hat{U}_{N-2,N-1} \\ & & & & \hat{U}_{N-1,N}. \end{pmatrix} \tag{2.1}$$

where $\Delta t' = \Delta t/(\text{no. of gates})$. Again it is important to have the transposed sweep to maintain the orthogonality and have 2nd order. The operations were applied in an optimized way such that least number of SWAP gates were applied [15].

3 Simulation

For the TEBD code the following Hamiltonian was simulated,

$$\frac{H_{\text{Ryd}}(t)}{\hbar} = \frac{\Omega(t)}{2}\sum_i \sigma_i^x - \Delta(t)\sum_i n_i + V_{nn}\sum_{i,j\in\mathcal{N}(i)} n_i n_j + V_d \sum_{i,j\in\mathcal{N}_d(i)} n_i n_j + V_h \sum_{i,j\in\mathcal{N}_h(i)} n_i n_j \quad (3.1)$$

where $\mathcal{N}(i)$, $\mathcal{N}_d(i)$, $\mathcal{N}_h(i)$ corresponds to the set of nearest, diagonal next-to-nearest, horizontal next-to-nearest neighbors respectively. The index j runs only over the forward neighbors to ensure double counting does not take place. Ω is the coupling to the Rydberg state (Rabi frequency), Δ is the detuning, and V_{nn}, V_d, V_h are the strength of the Rydberg interaction (V_0). The Pauli matrices are σ_i^α and $n_i = (1-\sigma_i^z)/2$. The values of the parameters are as follows: $V_{nn} = 12.5\ \Omega_{max}$, $V_h = 0.4\ \Omega_{max}$, $V_d = V_h/2.37$. Ω_{max} being the normalizing coefficient is kept as 1 during the simulation. The parameters $\Omega(t)$ and $\Delta(t)$ have the following time dependence (fig. 5):

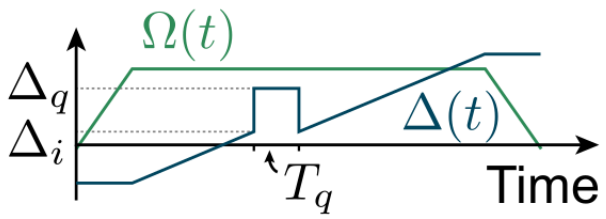
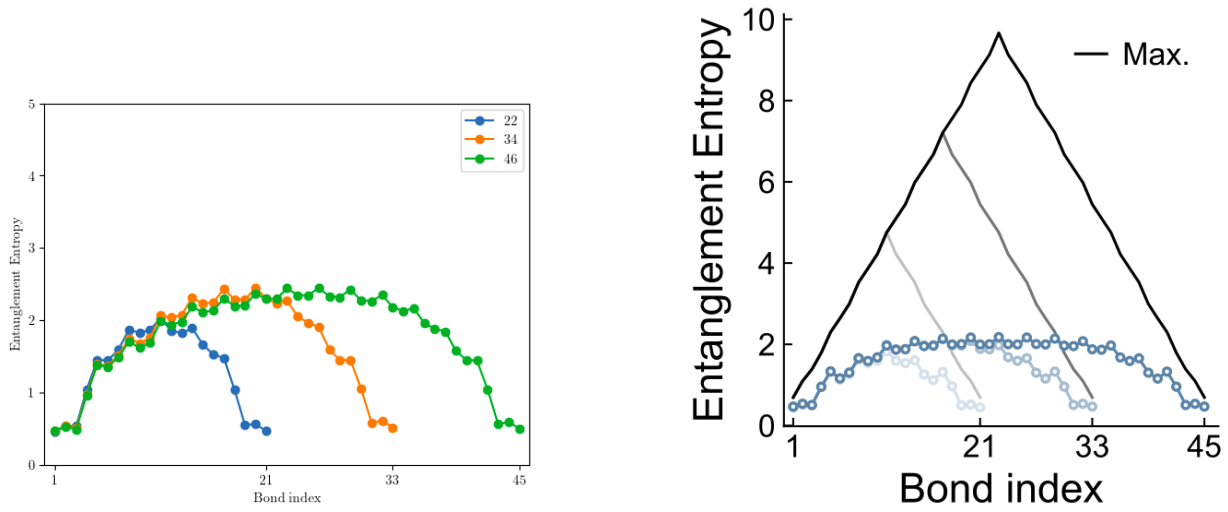


Figure 5: Functional dependence of $\Omega(t)$ and $\Delta(t)$ with time. The parameter Δ runs from $-4\ \Omega_{max}$ to $10\ \Omega_{max}$. The slope during the sweep is kept at $\dot{\Delta} = R_0$, where $R_0 = \Omega_{max}^2/(2\pi)$. The slope for $\Omega(t)$ at the edges is also kept as R_0. The quench parameters are fixed at $\Delta_i = 0.55\ \Omega_{max}$, $\Delta_q = 1.75\ \Omega_{max}$, and $T_q = 0.45\ \Omega_{max}$ (which is the optimal choice of the quench duration).

4 Results

Fig. 4 of the paper has been successfully reproduced: fig. 6

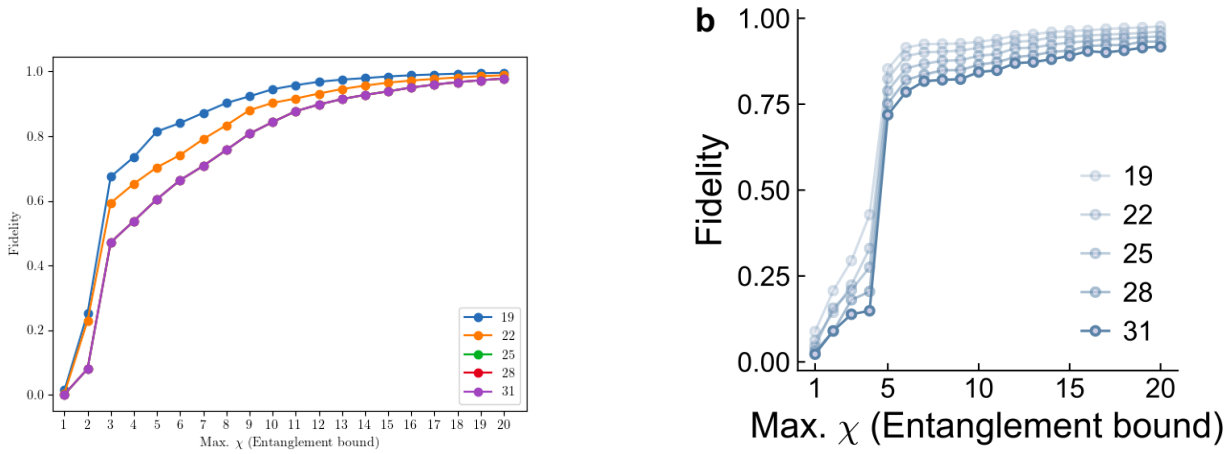


(a) Custom TEBD simulation

(b) Reference from the paper

Figure 6: The von Neumann entanglement entropy at the end of the SQS protocol, indicating the absence of volume-law entanglement generation. The dynamics are simulated using tensor-network methods for the Rydberg Hamiltonian (N = 22, 34, 46) including up to NNN interactions.

The black lines from the reference in fig. 6b correspond to the entanglement entropy of PXP model. They were most likely derived theocratically rather than through simulation because of the complexity of 3 site terms of the Hamiltonian and the quasi structure.



(a) Custom TEBD simulation

(b) Reference from the paper

Figure 7: TEBD simulation of the SQS protocol for various small bond dimensions χ, effectively restricting the dynamics to a Hilbert space with bounded entanglement. As a measure of the fidelity of the simulation, we show the norm of the matrix product state following truncation to a fixed bond dimension.

The discrepancy between the reference and the custom TEBD simulation in fig. 7 could most likely be because of the way the unfolding the quasi 1D lattice was done. For example, if

the NN sites were kept far away, a higher bond dimesnion will be necessary to have complete state description. The paper does not mention which type of unfoalding technique they use (or if they use some other technique all together).

The code was also tested against Fig. 11 of the paper fig. 8 and fig. 9. For the custom simulation of the SQS protocol, we see an extra oscillation section 4, which is most likely because the exact parameters were not mentioned for the given simulation and were chosen heuristically.

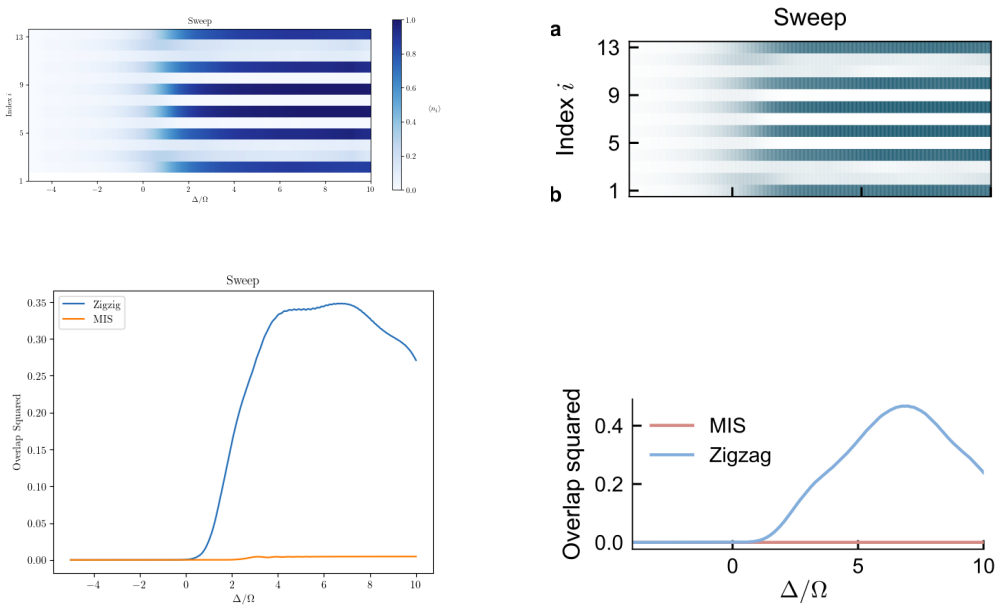


Figure 8: Comparison of the Sweep protocol between custom TEBD simulation (left) and reference from the paper (right).

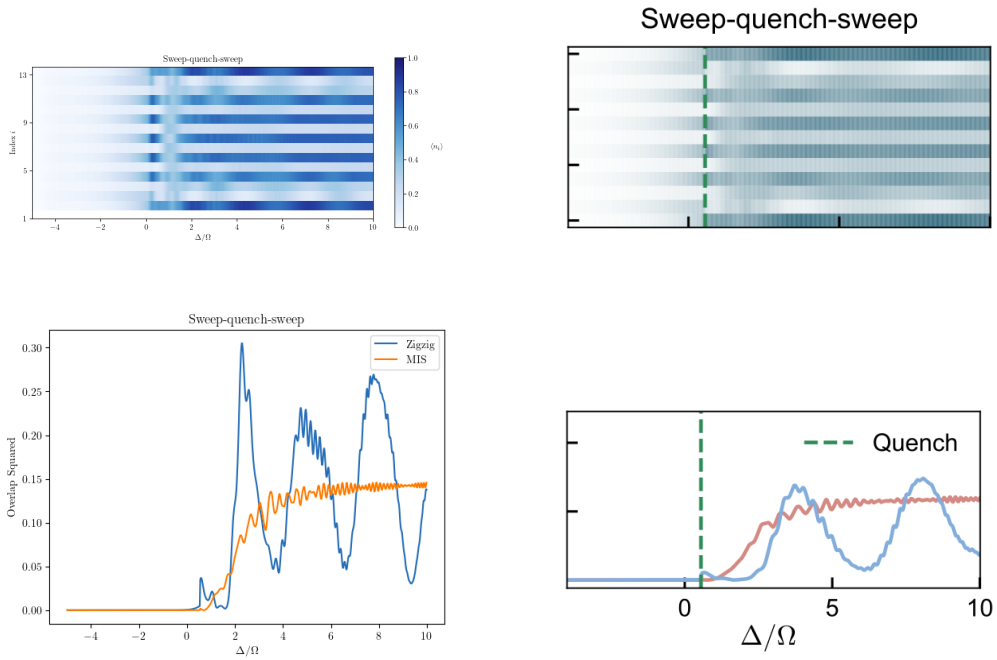


Figure 9: Comparison of the SQS protocol between custom TEBD simulation (left) and reference from the paper (right). The dotted green line represents the quench

5 Remarks

The simulation was also done for bang-bang protocol (quenching the system again and again for a given interval) which did not better results.

References

[1] S. Balasubramanian, Shuyang Han, B. T. Yoshimura, and J. K. Freericks. Bang-bang shortcut to adiabaticity in trapped-ion quantum simulators. *Physical Review A*, 97(2), February 2018.

[2] Jacob C Bridgeman and Christopher T Chubb. Hand-waving and interpretive dance: an introductory course on tensor networks. *Journal of Physics A: Mathematical and Theoretical*, 50(22):223001, May 2017.

[3] Glen Evenbly. Tensors.net. https://www.tensors.net/. Website maintained by Glen Evenbly (www.glenevenbly.com). Email: contact@tensors.net.

[4] Github. Tensor network methods with julia. https://quantumghent.github.io/TensorTutorials/.

[5] Github. Time-evolving block-decimation (tebd). https://tensornetwork.org/mps/algorithms/timeevo/tebd.html.

[6] Johannes Hauschild and Frank Pollmann. Efficient numerical simulations with tensor networks tensor network python (tenpy). *SciPost Physics Lecture Notes*, October 2018.

[7] Ana Hudomal, Jean-Yves Desaules, Bhaskar Mukherjee, Guo-Xian Su, Jad C. Halimeh, and Zlatko Papić. Driving quantum many-body scars in the pxp model. *Physical Review B*, 106(10), September 2022.

[8] Jhauschild. Jhauschild lecture comp methods tutorials for the lecture computational methods in many-body physics. https://github.com/jhauschild/lecture_comp_methods/tree/master?tab=readme-ov-file.

[9] Korbinian Kottmann. Introducing matrix product states for quantum practitioners. https://pennylane.ai/qml/demos/tutorial_mps, Nov 2024.

[10] Igor Lesanovsky and Hosho Katsura. Interacting fibonacci anyons in a rydberg gas. *Physical Review A*, 86(4), October 2012.

[11] Erich Mueller. Tensor network approaches to solving the many-body problem: Basic training in condensed matter physics. https://drive.google.com/file/d/15NLtvSJ7IkoHyw0keNMyo5pqQLtEWXz2/view, 2020. Lecture Notes.

[12] Johann Ostmeyer. Optimised trotter decompositions for classical and quantum computing. *Journal of Physics A: Mathematical and Theoretical*, 56(28), 2023.

[13] Sebastian Paeckel, Thomas Köhler, Andreas Swoboda, Salvatore R. Manmana, Ulrich Schollwöck, and Claudius Hubig. Time-evolution methods for matrix-product states. *Annals of Physics*, 411:167998, 2019.

[14] Robert N. C. Pfeifer, Glen Evenbly, Sukhwinder Singh, and Guifre Vidal. Ncon: A tensor network contractor for matlab. https://arxiv.org/abs/1402.0939, 2015.

[15] Johannes Schachenmayer. Dynamics and long-range interactions in 1d quantum systems. Technical report, Physik Department der Technischen Universität München, Munich, Germany, November 2008. Durchgeführt am Institut für Theoretische Physik der Universität Innsbruck.

[16] Paul Secular. Classical verification of a quantum simulator: local relaxation of a 1d bose gas. https://arxiv.org/abs/2401.05301, 2024.

[17] A. T. Sornborger and E. D. Stewart. Higher-order methods for simulations on quantum computers. *Phys. Rev. A*, 60:1956–1965, Sep 1999.

[18] E M Stoudenmire and Steven R White. Minimally entangled typical thermal state algorithms. *New Journal of Physics*, 12(5):055026, May 2010.

[19] Miles Stoudenmire. Tutorial on tensor networks and quantum computing with miles stoudenmire. https://www.youtube.com/watch?v=fq3_7vBcj3g.

[20] Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. *Physical Review Letters*, 91(14), October 2003.

[21] Guifré Vidal. Efficient simulation of one-dimensional quantum many-body systems. *Physical Review Letters*, 93(4), July 2004.

[22] Jan von Delft. Tensor networks 2022 – overview. https://www2.physik.uni-muenchen.de/lehre/vorlesungen/sose_22/tensor_networks_22/skript/index.html.

[23] Wikipedia. "time-evolving block decimation". https://en.wikipedia.org/wiki/Time-evolving_block_decimation, Jul 2024.