Infinite matrix product states applied to the transverse-field quantum Ising chain

Juan José Rodríguez & Oscar van Alphen

Advanced Numerical Methods in Many Body Physics
University of Amsterdam

June 28, 2021

Model: Infinite transverse-field Ising chain

$$H = -J \sum_{\langle i,j \rangle}^{\infty} \sigma_i^z \sigma_j^z - \Gamma \sum_{i}^{\infty} \sigma_i^x$$

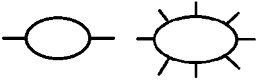
- $J > \Gamma$: Ordered phase
- $J = \Gamma$: Phase transition
- $J < \Gamma$: Disordered phase
- Order parameter: $m = \langle \frac{1}{N} | \sum_i^{\infty} \sigma_i^z | \rangle$
- Translationally invariant
- Chain is infinite: straight into the thermodynamic limit.

Introduction to tensor (networks)

1. The N-particle wave function: $|\Psi\rangle = \sum_{i_1,i_2,\ldots,i_N} \Psi_{i_1 i_2 \ldots i_N} |i_1 i_2 \ldots i_N\rangle$ The coefficients $\{\Psi_{i_1,i_2,\ldots,i_N}\}$ can be seen as either elements of a size $(2S+1)^N$ array or as the elements of a rank-N tensor.
2. Simple example, N = 2:

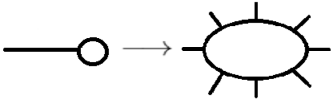
$$|\Psi\rangle = \Psi_{00}|00\rangle + \Psi_{01}|01\rangle + \Psi_{00}|10\rangle + \Psi_{11}|11\rangle \in \mathcal{H}^2$$
$$\longrightarrow \begin{bmatrix} \Psi_{00} & \Psi_{01} \\ \Psi_{10} & \Psi_{11} \end{bmatrix} \in \mathbb{C}^{2\times 2}$$

3. For N > 2 it becomes convenient to use tensor diagrammatic notation



Introduction to tensor networks

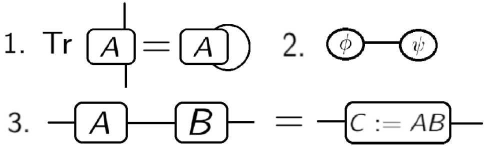
8-particle wave function



Left: Column vector (rank-1 tensor): a 1 dimensional array with $(2S+1)^8$ coefficients.
Right: 8-dimensional tensor (rank-8 tensor): A $(2S+1) \times \ldots_{8 times} \times (2S+1)$ array.

Introduction to tensor networks

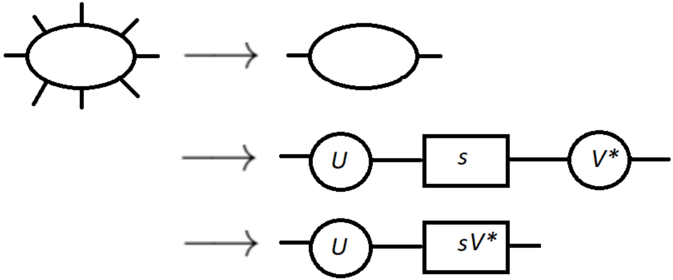
- Tensor networks: A reformulation of (multiparticle) quantum mechanics
- Connected lines denote contractions: any sum of (products of) tensor coefficients.
- Examples:
  1. Trace: $\operatorname{Tr} A = \sum_i A_{ii}$
  2. Inner products: $\langle\phi|\psi\rangle$
  3. Matrix product: $\sum_j A_{ij} B_{jk} = C_{ik}$



Introduction to tensor networks

- In stead of using a rank-N tensor with $(2S+1)^N$ coefficients, we make an Ansatz in the form of a tensor network, i.e. a network of tensors with smaller ranks connected by lines, where the coefficients scale polynomially in N.
- How is this possible?
- Low energy states obey an entanglement area law

Introduction to tensor networks



$|\Psi\rangle = \sum_{i,j} \Psi_{ij} |i\rangle |j\rangle = \sum_k s_k |a_k\rangle |b_k\rangle$

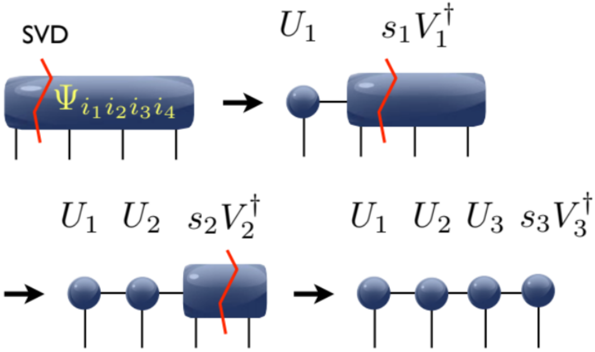
$S = -\sum_s s_k^2 log\, s_k^2$

Random state: $S \sim L^d$

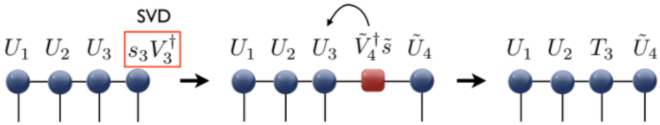
Low energy states: $S \sim L^{d-1}$

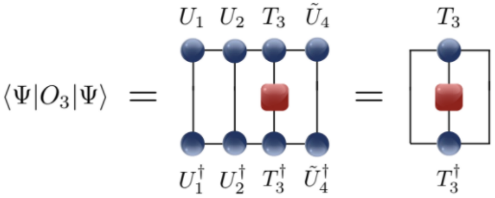
Critical ground states: $S \sim log\, L$

MPS decomposition



Canonical form of an MPS

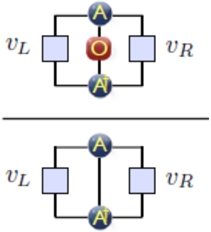




Measuring the energy / magnetization

- Once in canonical form, it is very easy to measure observables
- Contract the MPS with a tensor representing the quantity to measure (hamiltonian, spin operator, etc)

$$\frac{\langle\Psi|\hat{O}|\Psi\rangle}{\langle\Psi|\Psi\rangle} =$$



The algorithm

Four main components:

1. Build the starting MPS ansatz at random
2. Apply imaginary-time evolution gates
3. Bring the MPS to canonical form
4. Measure observables

Building the starting (random) ansatz

The Ansatz of the iTEBD algorithm:

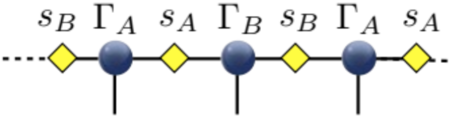


Figure 1: P. Corboz (2018)

Initialize random $\chi \times 2 \times \chi$ tensor Γ_A for even numbered sites, and a different one, Γ_B for odd numbered sites. Similarly, random diagonal matrix λ_A and λ_B

Imaginary-time evolution gates

- Main ingredient: applying imaginary-time evolution gates on the MPS
- Alternative form of the power method:

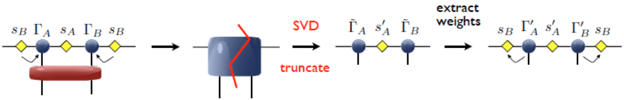
$$e^{-\beta \hat{H}} \left|\psi_{init}\right\rangle \rightarrow \left|\psi_{GS}\right\rangle, \quad \beta \rightarrow \infty$$

- Combine with Trotter-Suzuki decomposition

$$e^{-\beta \hat{H}} = \left(e^{-\tau \hat{H}}\right)^{M} \rightarrow e^{-\tau \sum_b \hat{H}_b} \approx \prod_b e^{-\tau \hat{H}_b}$$

Applying imaginary-time evolution gates

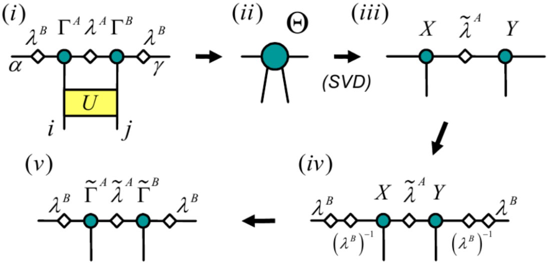
- To apply the time-evolution gates $e^{-\tau \hat{H}}$, contract MPS
- Apply SVD afterwards to maintain in canonical form
- Important to truncate the smallest singular values / keep the D largest singular values (also for numerical stability)
- Extract the weights at the end to keep original ansatz



Applying imaginary-time evolution gates

Original formulation by Guifré Vidal in 2007:

Figure 2: G.Vidal (2007)



Bring the MPS to canonical form

We can obtain the canonical form by two alternative ways:

- Applying several thermalization imaginary-time evolution gates before starting to measure.
- Computing the dominant eigenvectors σ and μ of the transfer operators (doesn't need thermalization).

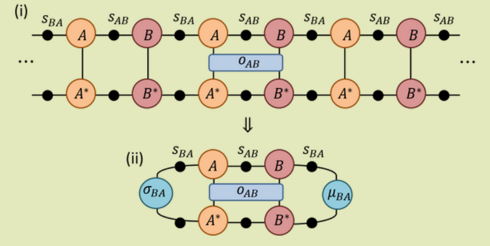


Figure 3: G. Evenbly (2020)

Bring the MPS to canonical form

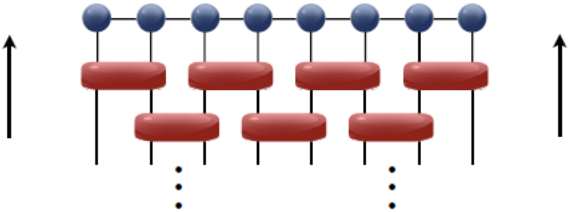
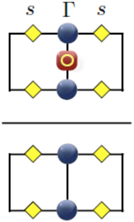


Figure 4: P. Corboz (2018)

Measuring observables (energy and magnetization)

Once in canonical form it is very easy to measure observables: just contract the tensor representing the observable ($\hat{H}$, $\sigma_z \bigotimes I$, etc.) with two MPS as explained before



Optimizing the algorithm

- As seen before: Important to trim the smallest singular values when applying evolution gates.
- Sometimes convergence is very slow: wait until $\Delta E < tol$ instead of a fixed number of iterations.
- Better to run simulations sequentially instead of always from random (although more numerical instability!).
- The order parameters converges much slower than the energy. Important for precise results.
- Useful trick: second-order Trotter scheme.

Finite size scaling

- At the critical point, the finite bond dimension acts as a finite size
- We can choose the critical correlation length as the finite size of the system

$$L \equiv \xi_D(\Gamma = \Gamma_c)$$

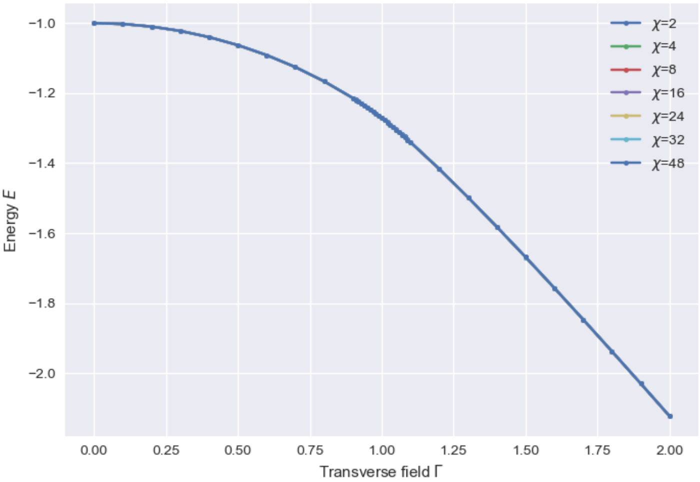
- The different observables behave as a power law of the finite size. Example: magnetization

$$m(\Gamma = \Gamma_c, L) \propto \xi_D^{-\beta/\nu}$$

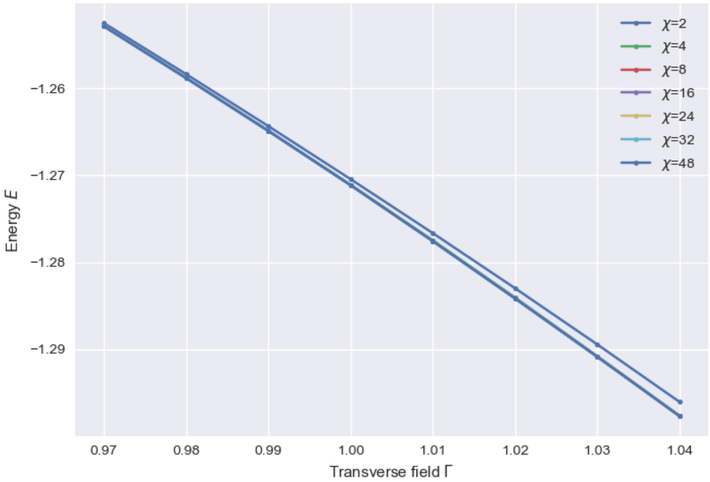
- This allows to fit for the critical exponents

$$\log m \propto \log \xi_D$$

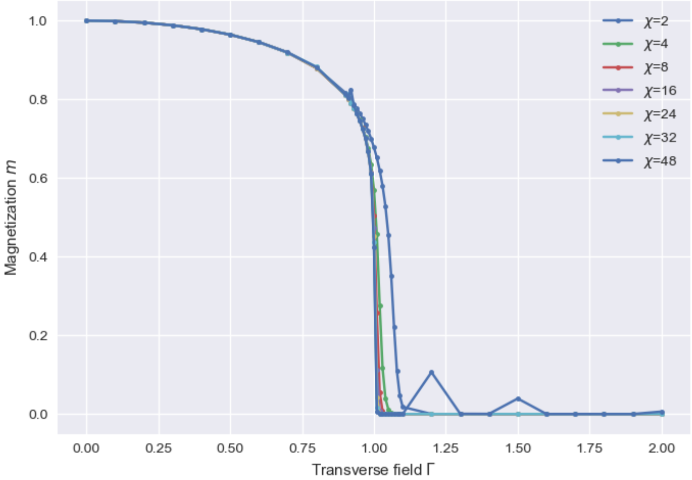
Dependence on transverse field Γ



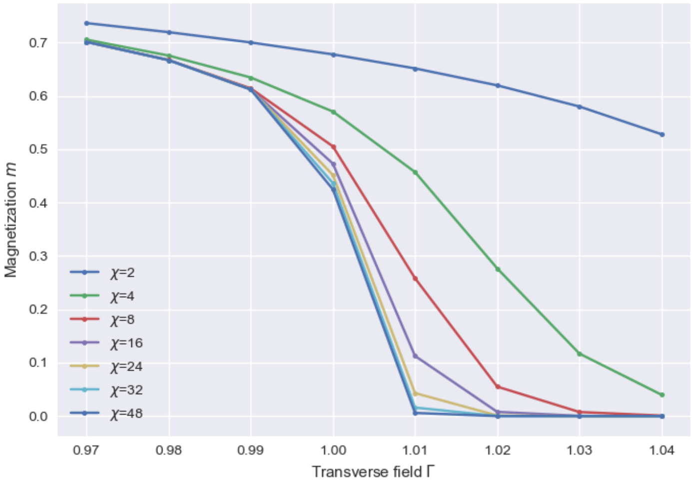
Dependence on transverse field Γ



Dependence on transverse field Γ

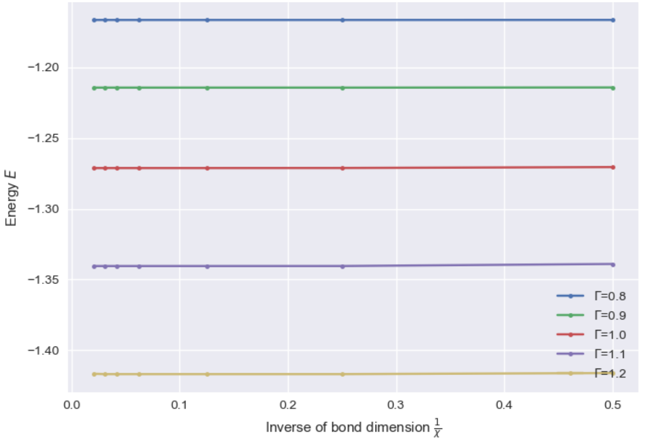


Dependence on transverse field Γ

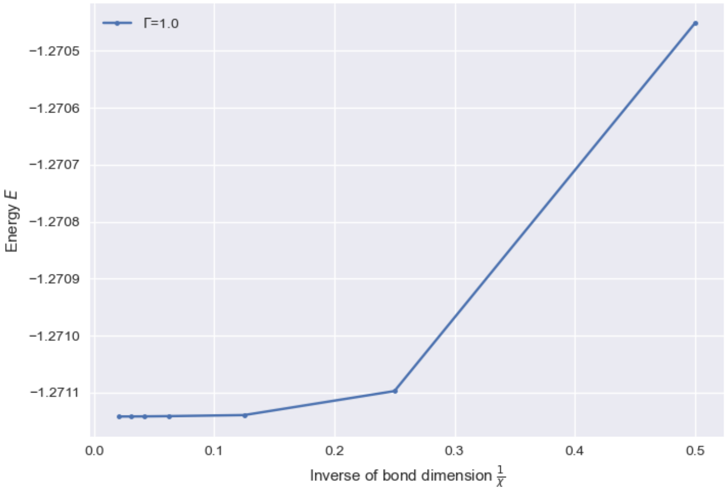


Dependence on bond dimension χ

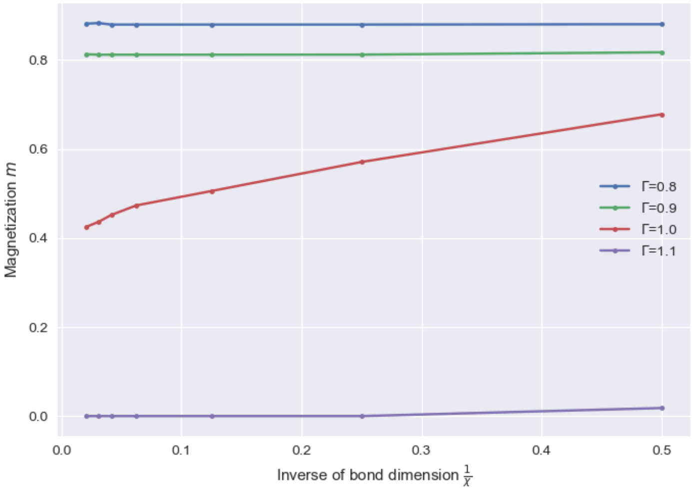
Variational theorem: $E_\psi \geq E_0$



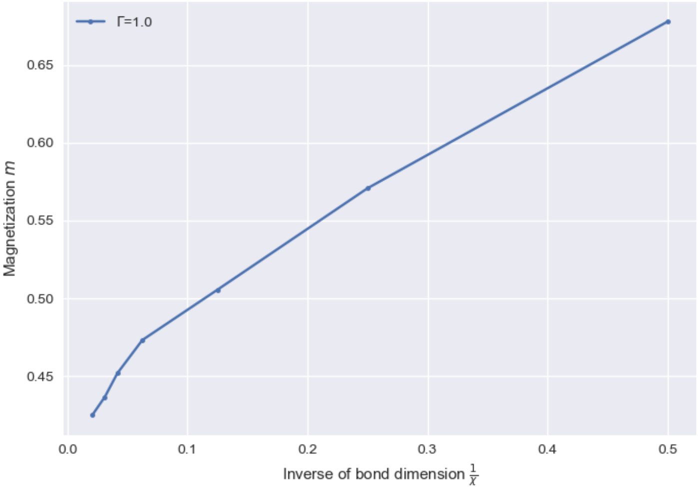
Dependence on bond dimension χ



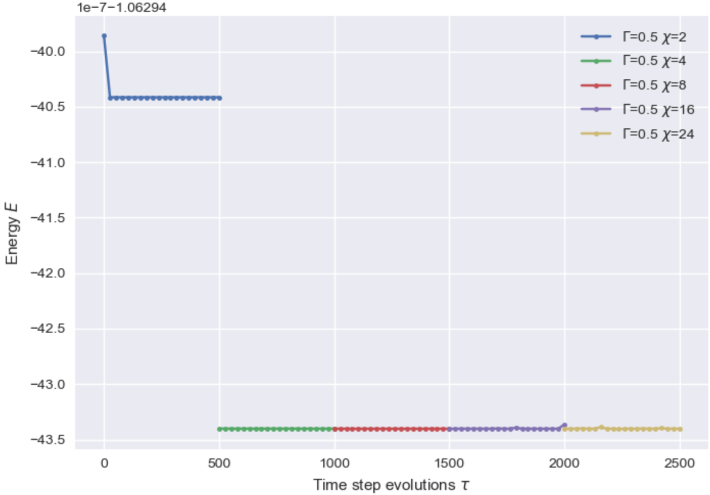
Dependence on bond dimension χ



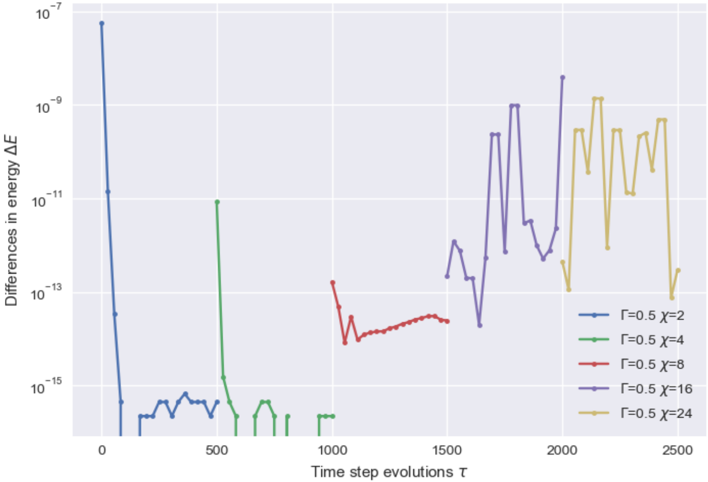
Dependence on bond dimension χ



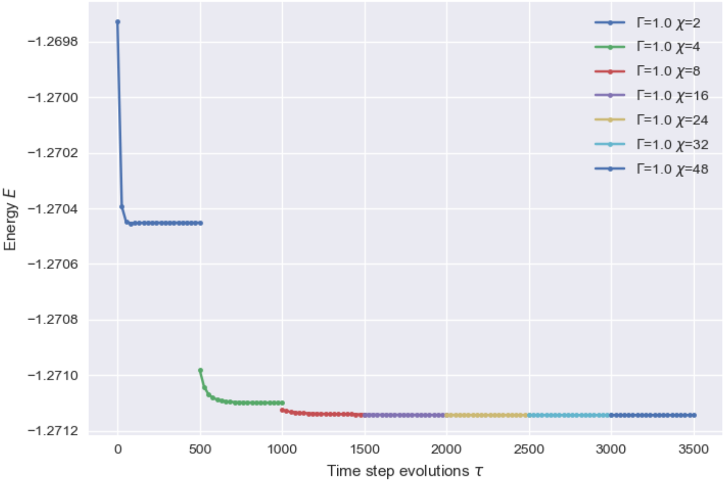
Dependence on imaginary time-evolution steps: far from critical



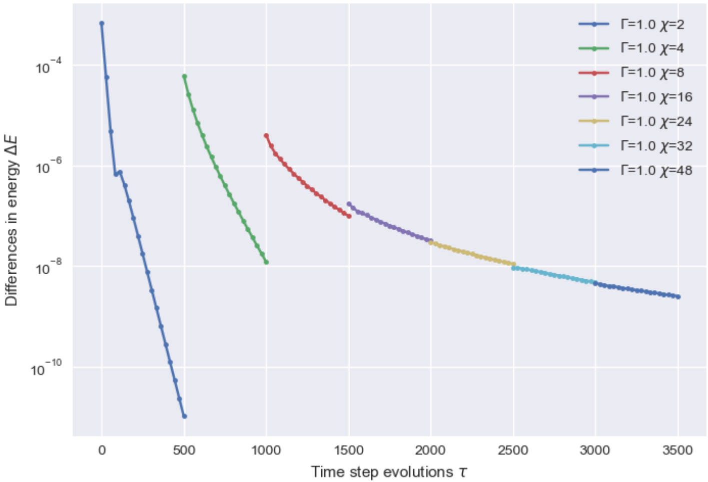
Dependence on imaginary time-evolution steps: far from critical



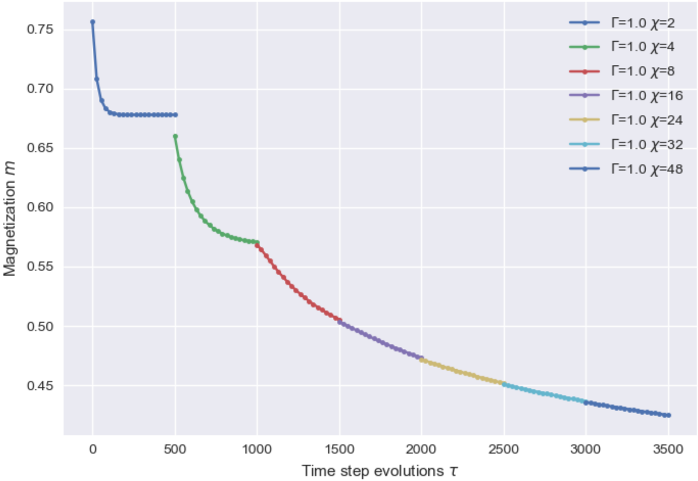
Dependence on imaginary time-evolution steps: critical point



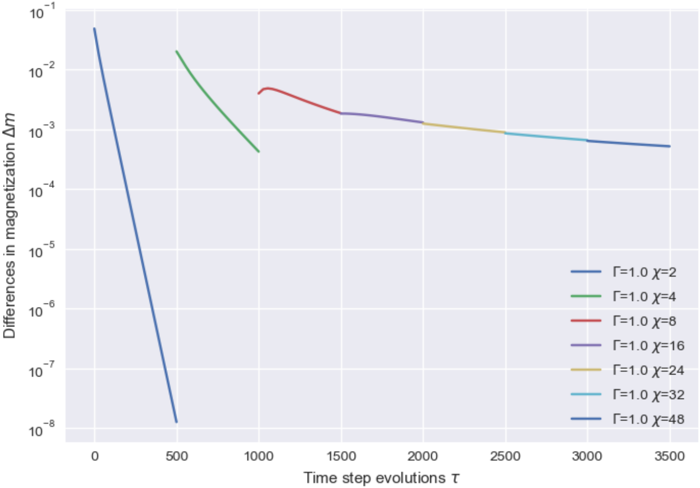
Dependence on imaginary time-evolution steps: critical point



Dependence on imaginary time-evolution steps: critical point



Dependence on imaginary time-evolution steps: critical point



Some other computations

- Entanglement entropy:

$$S_A = -\operatorname{Tr} \hat{\rho}_A \log \hat{\rho}_A = -\sum_k p_k \log p_k$$

- Correlation length (directly from transfer matrix): $\frac{1}{\xi} = \log \frac{\Lambda_0}{\Lambda_1}$
- Finite size scaling:

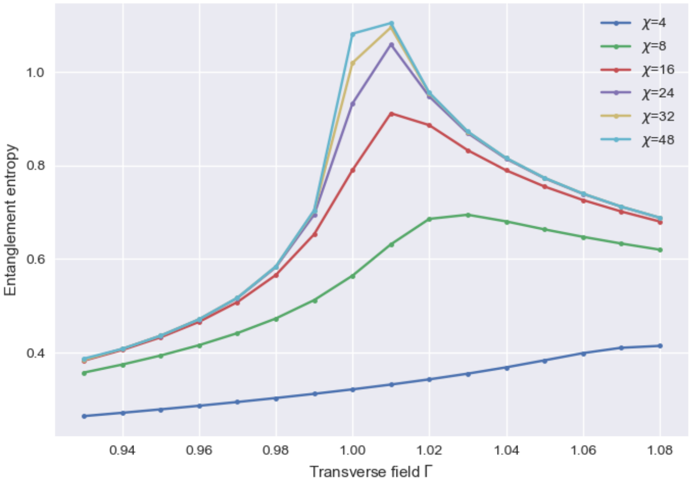
$$m(\Gamma = \Gamma_c, L) \propto L^{-\beta/\nu} \to \xi_D^{-\beta/\nu}$$

- Data collapse:

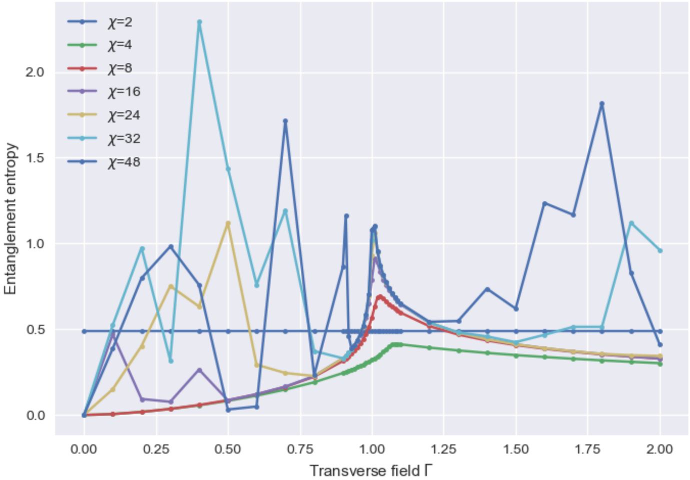
$$m(t, L) = L^{-\frac{\beta}{\nu}} \mathcal{F}(t \cdot L^{\frac{1}{\nu}})$$

$$x \equiv t \cdot L^{\frac{1}{\nu}} \to \mathcal{F}(x) = mL^{\beta/\nu}$$

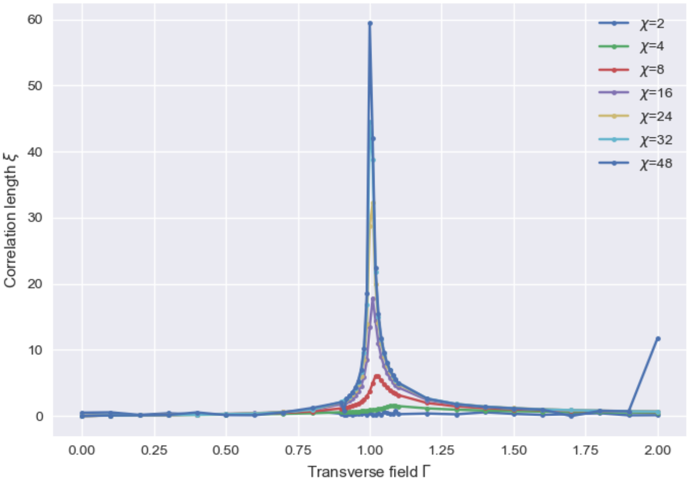
Entanglement entropy



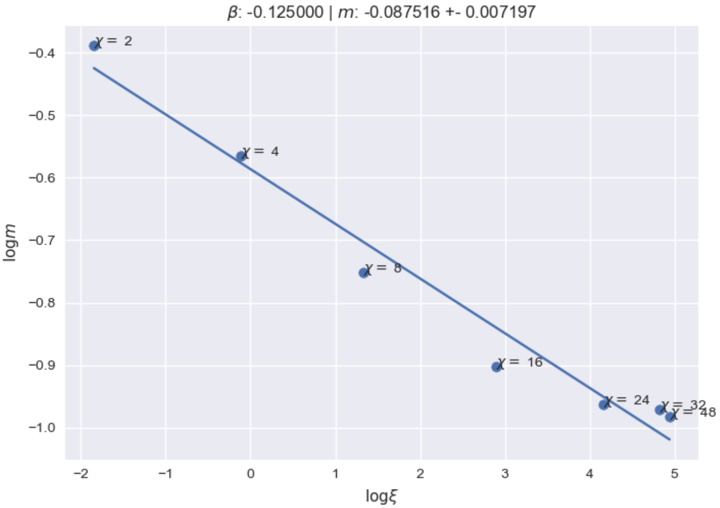
Entanglement entropy



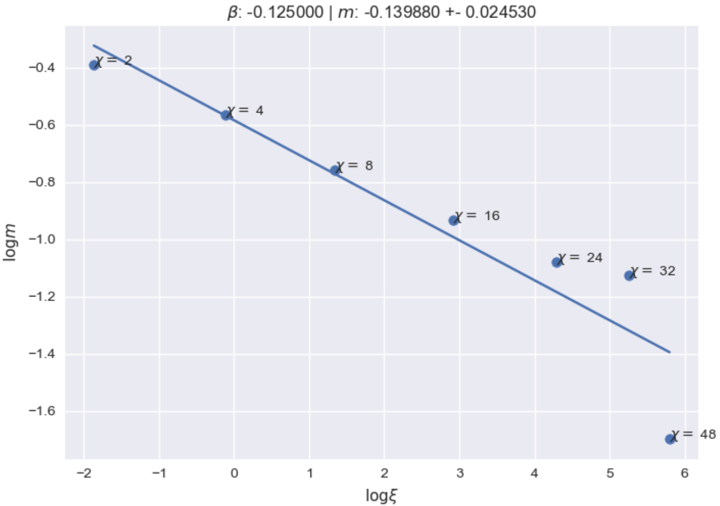
Correlation length



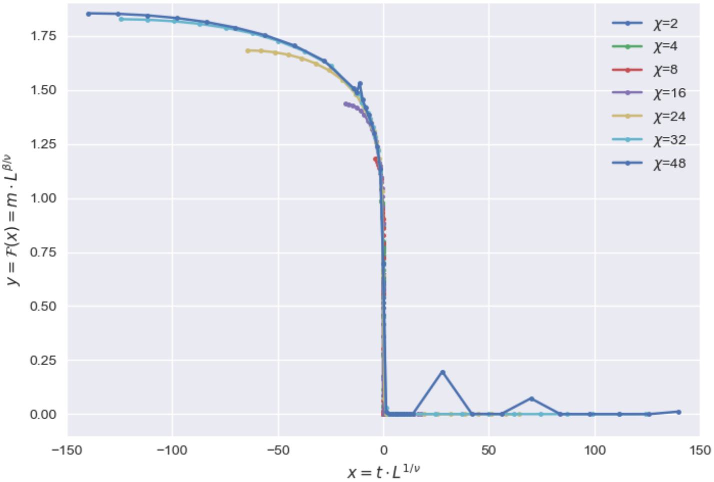
Critical exponents ($\Delta E < 10^{-10}$)



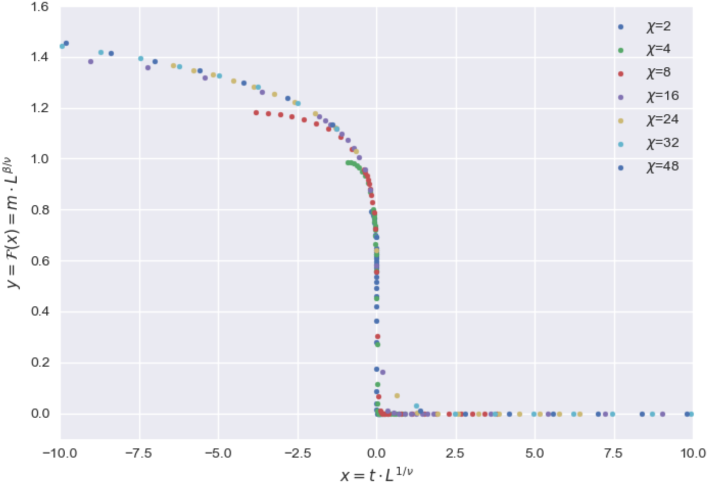
Critical exponents ($\Delta m < 10^{-5}$)



Data collapse



Data collapse



Conclusion

Tensor networks pose a lot of advantages

- Computes quantum systems in a quasi-exact way without the need of Monte Carlo methods (negative sign problem).
- Can obtain results to arbitrarily high (machine-level) precision in some circumstances for moderatelly small χ, even at the critical point.
- Can be very easily adapted to other quantum systems, or with more effort to higher dimensions (PEPS).

We have obtained reasonably good results even at the critical point with modest computational resources!

Conclusion

We found a number of difficulties:

- Very long convergence times at the critical point.
- Numerical instabilities at large χ (can be circumvented).
- Need of very precise values to compute some properties (eg: critical exponents, data collapse).

Approach left open:

- Try to run simulations for varying time-steps τ to try to speed up and increase accuracy

1. G. Vidal (2007)
2. P. Corboz (2018)
3. G. Evenbly (2020)

Thanks!