Numerical approaches to quantum many-body non-equilibrium

Université de Strasbourg

CNRS — dépasser les frontières

CESQ — CENTRE EUROPÉEN DE SCIENCES QUANTIQUES

www.schachenmayer.fr

www.cesq.eu

Johannes Schachenmayer: schachenmayer@unistra.fr

Numerical approaches to quantum many-body non-equilibrium

Goal:

A tour through some numerical methods for simulating large quantum many-body non-equilibrium dynamics, with examples. Learning physics by simulating it.

Lecture 1: Foundations (QM on a computer), Runge-Kutta, Applications to ultra-cold bosonic systems

Lecture 2: Spin-model physics, Krylov space approaches, Open system methods

Lecture 3: Large systems: Matrix Product States (DMRG), Applications to spin-models and Bose-Hubbard

- Some text recommendations:

  - *Numerical recipes - The Art of Scientific Computing (a classic), on-line: https://numerical.recipes*
  - *General references to openly available publications in class*

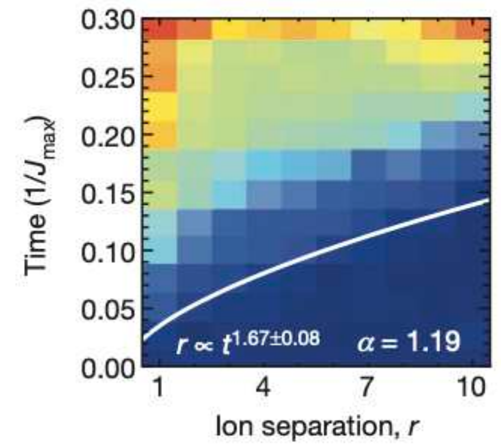
- Language recommendation (used for examples): Julia, https://julialang.org/ (open source, easy, fast linear algebra)

- What these lectures are **not:**

  - *Complete: Many techniques are not discussed (e.g. Monte-Carlo, Fermions, Phase space methods, …)*
  - *Computer science class: No proofs of complexity etc.*
  - *Numerical tutorial: There will be code snippets … incentive to do it yourself*

Last time

- We discussed some general spin-model physics with long-range couplings, as they can e.g. be engineered in trapped ion systems. We discussed how to construct Hamiltonians using Kronecker products. It's very important to use **sparse matrices**.

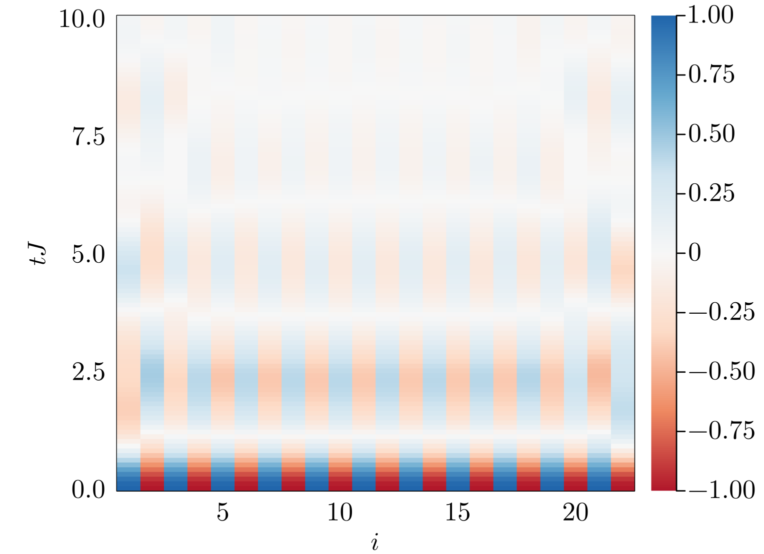


$$\hat{\sigma}_i^- = \begin{pmatrix} 1 & & \\ & 1 & \\ & & \ddots \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & & \\ & 1 & \\ & & \ddots \end{pmatrix}$$

- We introduced a new time-evolution algorithm for linear systems, based on: **Krylov space**. Krylov space is a vector-space constructed from an initial state and the evolution matrix:

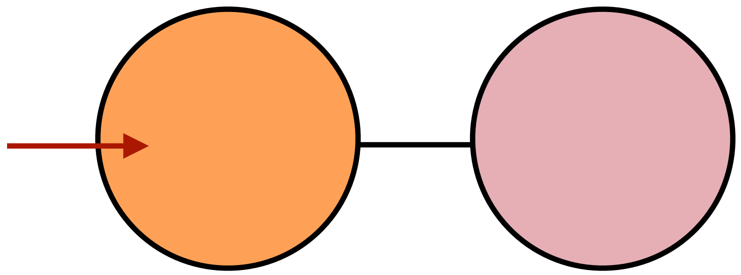
$$\text{span}\left(\hat{A}^0 \left|\psi_0\right\rangle, \hat{A}^1 \left|\psi_0\right\rangle, \hat{A}^2 \left|\psi_0\right\rangle, \ldots, \hat{A}^{m-1} \left|\psi_0\right\rangle\right)$$

Eigenvectors need to be made orthonormal, using **Arnoldi** or **Lanczos** iterations, then diagonalizations and matrix exponentials can be performed very efficiently on the much smaller **Krylov space.** This allows to easily simulate quantum dynamics of ~22 spins/qubits on a laptop.



- Finally we discussed how to simulate open system dynamics of **Lindblad master equations:**

$$\frac{d}{dt}\hat{\rho}_S = \mathcal{L}(\hat{\rho}_S) = -\mathrm{i}\left[\hat{H}_S, \hat{\rho}\right] + \sum_\eta \left(2\hat{L}_\eta \hat{\rho}_S \hat{L}_\eta^\dagger - \hat{L}_\eta^\dagger \hat{L}_\eta \hat{\rho}_S - \hat{\rho}_S \hat{L}_\eta^\dagger \hat{L}_\eta\right)$$

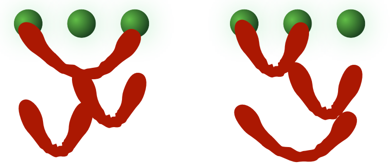


… either by **vectorization of the density matrix**, or by **quantum trajectories** (an unravelling of the density matrix into stochastic pure state evolutions)

Plan for today

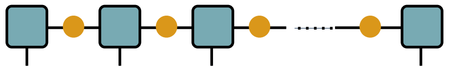
A strategy to simulate dynamics in systems way beyond 30 qubits - Matrix Product States

- Boring preliminaries: Suzuki-Trotter decomposition

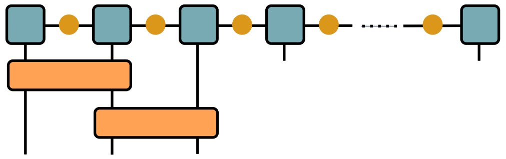


- Bipartite entanglement and singular value decompositions

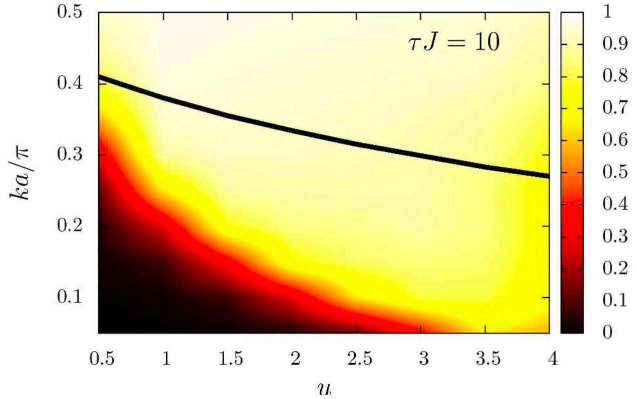
- Matrix product states (MPS)



- The time-evolving block decimation algorithm (TEBD)



- Application example of Bose-Hubbard and Spin-model dynamics

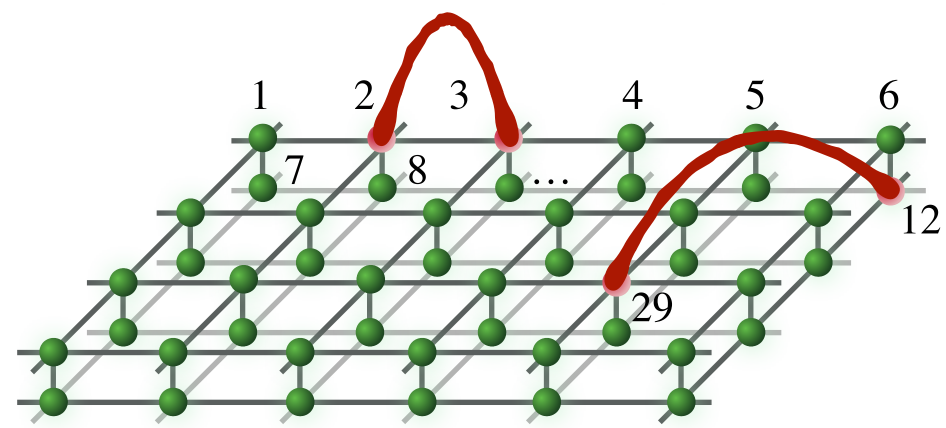


Lecture 3 - Trotter decompositions

$$e^{-\mathrm{i}t\hat{H}}|\psi_0\rangle = \,?$$

- **1.** Models with two-body “pair-wise” interactions

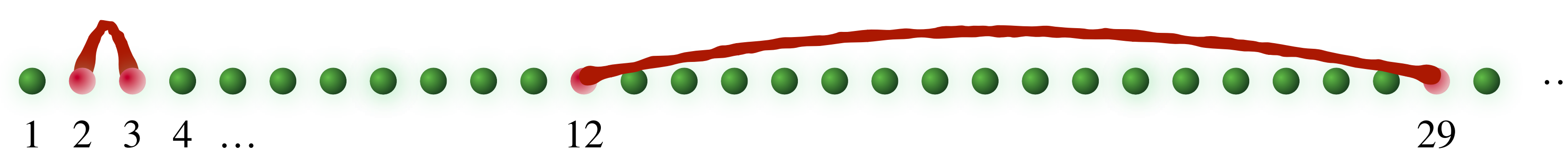
$$\hat{H} = \sum_{i>j} \hat{H}_{ij}$$



Example: Hopping $\hat{H} = -J \sum_{i}^{M-1} \left(\hat{b}_i^\dagger \hat{b}_{i+1} + \hat{b}_{i+1}^\dagger \hat{b}_i\right)$

- **2.** Typically interactions are “local” and decay as function of some distance (we don’t need to insist on this)

- **3.** W.l.o.g. let’s think about the system as one-dimensional



- **Note:** The ordering is really arbitrary, but depending on the type of interactions it can be “optimal”.
- **Note:** We can introduce operations that swap particles (like swap gates in a quantum computer) to decompose everything into interactions on nearest neighbors.

Lecture 3 - Trotter decompositions

$$\hat{H} = \sum_{i>j} \hat{H}_{ij}$$

$$\mathrm{e}^{-\mathrm{i}t\hat{H}} |\psi_0\rangle = ?$$

- The idea of a **Suzuki-Trotter decomposition**
- Let's evolve the system stroboscopically over small time steps: $\Delta t \ll \|\hat{H}_{ij}\|^{-1}$

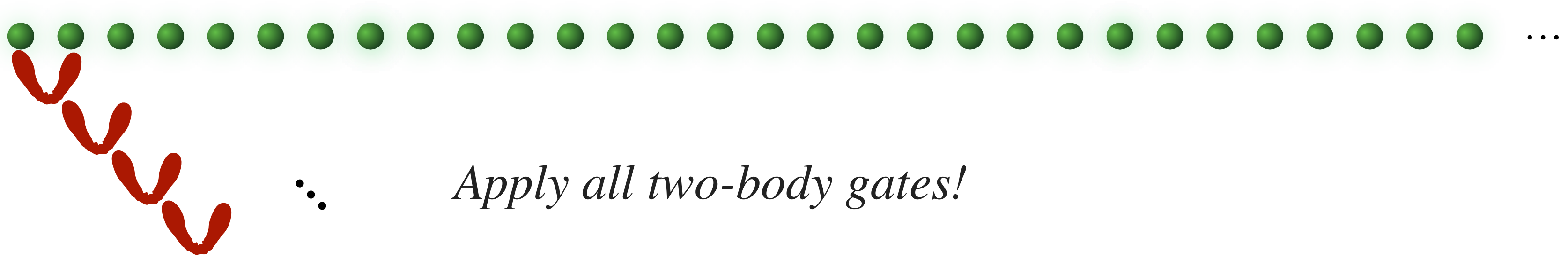
$$\mathrm{e}^{-\mathrm{i}\Delta t\hat{H}} = \mathrm{e}^{-\mathrm{i}\sum_{i>j}\Delta t\hat{H}_{ij}} = \prod_{i>j} \mathrm{e}^{-\mathrm{i}\Delta t\hat{H}_{ij}} + \text{error}$$

$$\text{error} \propto \Delta t^2, [\hat{H}_{ij}, \hat{H}_{jk}]$$

$$\mathrm{e}^{\alpha(\hat{X}+\hat{Y})} = \mathrm{e}^{\alpha\hat{X}}\mathrm{e}^{\alpha\hat{Y}}\mathrm{e}^{\frac{\alpha^2}{2}[\hat{X},\hat{Y}]+\mathcal{O}(\alpha^3)}$$

"Zassenhaus formula" Baker-Campbell-Hasudorf etc.

- For small enough time-step we can simulate the Hamiltonian evolution with "pair-wise gates"!



Apply all two-body gates!

- **Note:** Trotter decompositions are the key step to digital quantum simulations on a quantum computer.

S. Lloyd, Science 273 (1996)

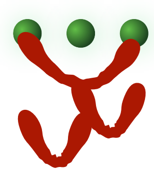
Lecture 3 - Trotter decompositions

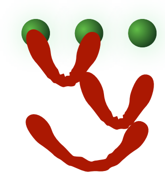
$$\hat{H} = \sum_{i>j} \hat{H}_{ij}$$

$$\mathrm{e}^{-\mathrm{i}t\hat{H}}|\psi_0\rangle = ?$$

- In practice: **Use a high order Trotter decomposition!**
- **For example:** Say we have a Hamiltonian $\hat{H} = \hat{H}_{12} + \hat{H}_{23} + \hat{H}_{13}$
- Then, one can show:

$$\mathrm{e}^{-\mathrm{i}\Delta t H} = \underbrace{\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{12}}\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{23}}\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{13}}}_{\textit{reversed sweep}\ \hat{U}^{\dagger}_{-\Delta t/2}}\ \underbrace{\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{13}}\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{23}}\mathrm{e}^{-\mathrm{i}\frac{\Delta t}{2}\hat{H}_{12}}}_{\textit{sweep}\ \hat{U}_{\Delta t/2}} + \mathcal{O}(\Delta t^3)$$





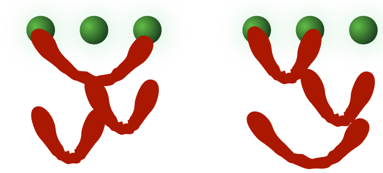
- **Observe:** **Twice** the number the sweeps, but error is now $\mathcal{O}(\Delta \boldsymbol{t^3})$
- Errors can be further minimized, in practice useful: **4th** order, then e.g. **18** sweeps and error $\mathcal{O}(\Delta \boldsymbol{t^5})$

see: A. T. Sornborger and E. D. Stewart, Phys. Rev. A 60, 1956 (1999)

Plan for today

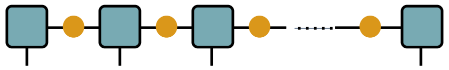
A strategy to simulate dynamics in systems way beyond 30 qubits - Matrix Product States

- Boring preliminaries: Suzuki-Trotter decomposition

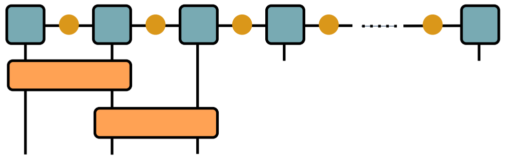


- Bipartite entanglement and singular value decompositions

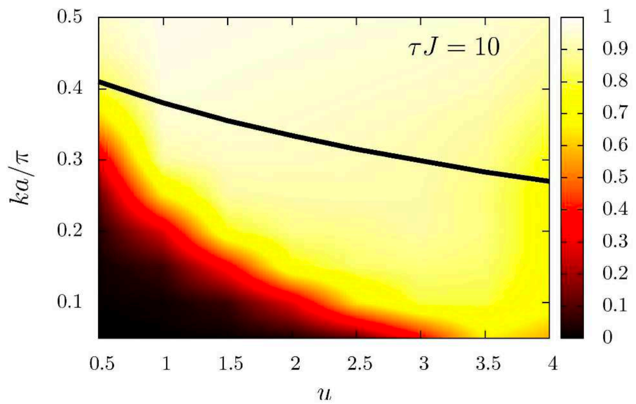
- Matrix product states (MPS)



- The time-evolving block decimation algorithm (TEBD)

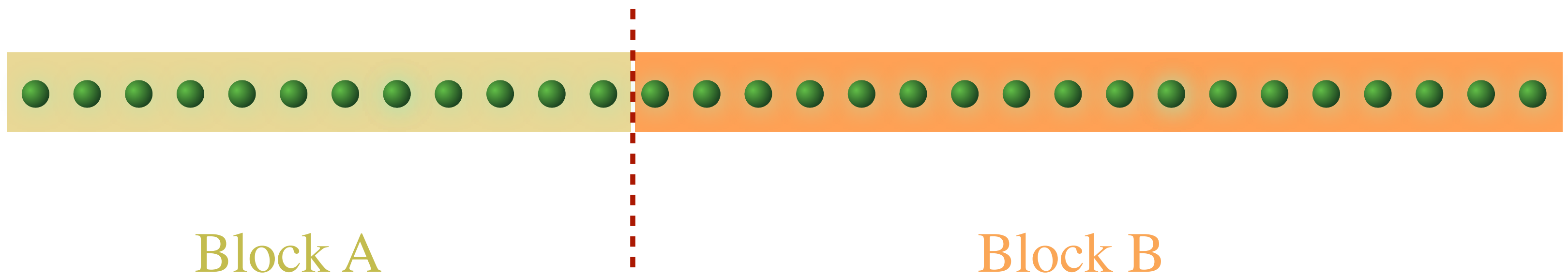


- Application example of Bose-Hubbard and Spin-model dynamics



Lecture 3 - Bipartite entanglement entropy

- Assume a pure state of our many-body system



$= |\psi\rangle$

- Bipartition:

- Definition: $|\psi\rangle \neq |\phi\rangle_A \, |\chi\rangle_B$ ⟷ **"Blocks A and B are entangled"**

- How to quantify the amount of entanglement?

State in block A

$$\rho_A = \mathrm{tr}_B \left(|\psi\rangle\langle\psi| \right)$$

State in block B

$$\rho_B = \mathrm{tr}_A \left(|\psi\rangle\langle\psi| \right)$$

- **Example:** Say the total state is a non-entangled product state $|\psi\rangle = |\phi\rangle_A \, |\chi\rangle_B$

… then:

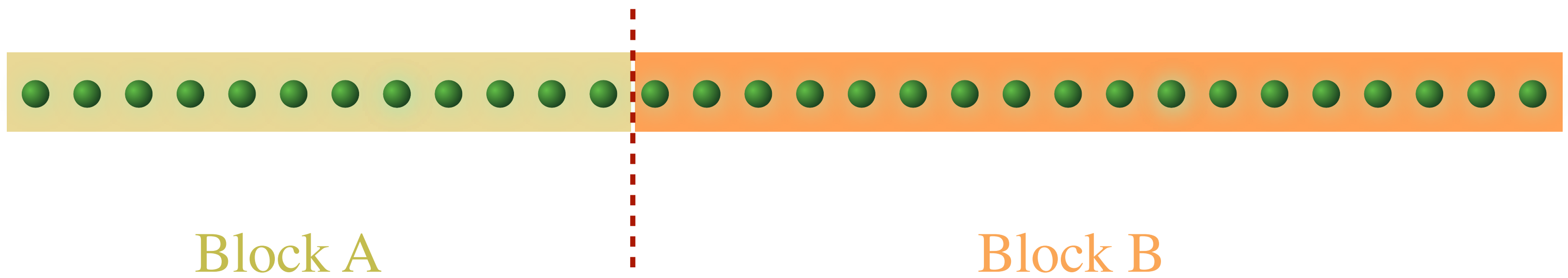
$$\rho_A = |\phi\rangle \langle\phi|_A \qquad \rho_B = |\chi\rangle \langle\chi|_B$$

Pure states!

The state of each block contains all information of the state in that block!

Lecture 3 - Bipartite entanglement entropy

- Assume a pure state of our many-body system



$$= |\psi\rangle$$

- Bipartition:

- Definition: $|\psi\rangle \neq |\phi\rangle_A \, |\chi\rangle_B$ ⟷ **"Blocks A and B are entangled"**

- **Example:** Say the total state of just two qubits is maximally entangled: $|\psi\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle_A \, |0\rangle_B + |1\rangle_A \, |1\rangle_B \right)$

…then: $\rho_A = \mathrm{tr}_B \left(|\psi\rangle\langle\psi| \right) = \langle 0|\psi\rangle\langle\psi|0\rangle_B + \langle 1|\psi\rangle\langle\psi|1\rangle_B$

$$|\psi\rangle\langle\psi| = \frac{1}{2} \left(|0\rangle_A |0\rangle_B \langle 0|_A \langle 0|_B + \cancel{|0\rangle_A |0\rangle_B \langle 1|_A \langle 1|_B} + \cancel{|1\rangle_A |1\rangle_B \langle 0|_A \langle 0|_B} + |1\rangle_A |1\rangle_B \langle 1|_A \langle 1|_B \right)$$

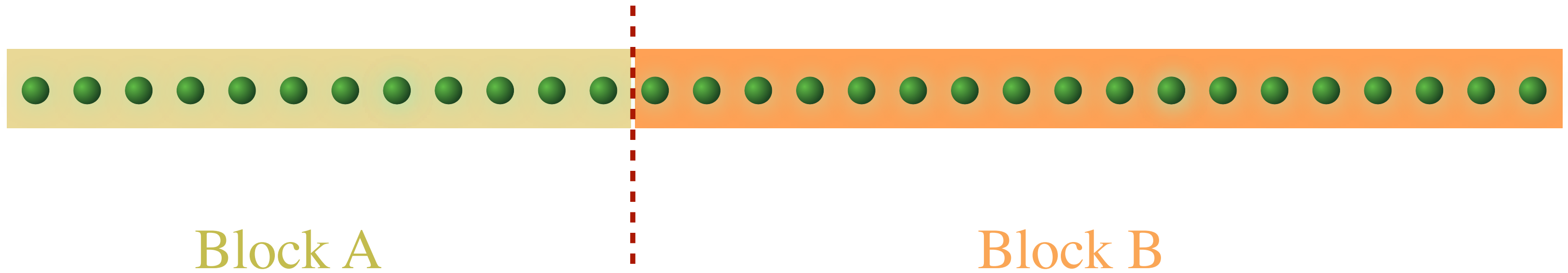
$$\rho_A = \frac{1}{2} \left(|0\rangle\langle 0|_A + |1\rangle\langle 1|_A \right) = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}_A$$

Idea: *Use the **entropy** of the reduced density matrix as entanglement measure!*

The state of a sub-block is a maximally mixed state!

Lecture 3 - Bipartite entanglement entropy

- Assume a pure state of our many-body system



$= |\psi\rangle$

- Bipartition:

- Definition: $|\psi\rangle \neq |\phi\rangle_A \, |\chi\rangle_B$ ⟷ **"Blocks A and B are entangled"**

- How to quantify the amount of entanglement?

State in block A

$$\rho_A = \mathrm{tr}_B \left(|\psi\rangle\langle\psi| \right)$$

State in block B

$$\rho_B = \mathrm{tr}_A \left(|\psi\rangle\langle\psi| \right)$$

- **von Neumann entanglement entropy:**

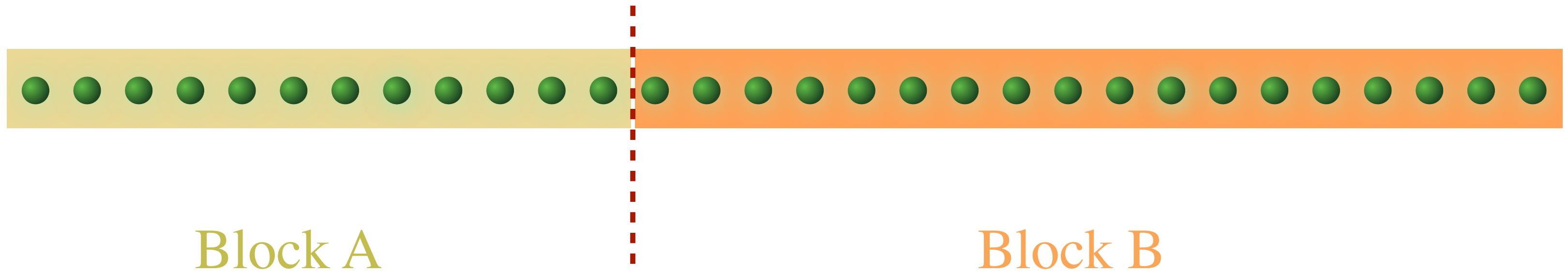
$$S_{\mathrm{vN}}(\rho_A) = -\mathrm{tr}_{\mathrm{A}} \left[\rho_A \log_2(\rho_A) \right] = -\sum_\alpha \lambda_\alpha \log_2(\lambda_\alpha) = S_{\mathrm{vN}}(\rho_B)$$

true for pure states on the full system

Eigenvalues of ρ_A

Lecture 3 - Bipartite entanglement entropy

- Assume a pure state of our many-body system



$= |\psi\rangle$

- Bipartition:

Block A   Block B

$d_A = \dim(\mathcal{H}_A)$ $w.l.o.g.\ d_A < \dim(\mathcal{H}_B) = d_B$

- Entanglement entropy:

$$S_{\mathrm{vN}}(\rho_A) = -\mathrm{tr}_{\mathrm{A}}\left[\rho_A \log_2(\rho_A)\right] = -\sum_\alpha \lambda_\alpha \log_2(\lambda_\alpha) = S_{\mathrm{vN}}(\rho_B)$$

***No** entanglement*

***Maximum** entanglement*

$\lambda_1 = 1, \lambda_{\alpha\neq 1} = 0$ $S_{\mathrm{vN}}(\rho_A) = 0$

$\lambda_\alpha = \frac{1}{d_A}$ $S_{\mathrm{vN}}(\rho_A) = \log_2(d_A)$

- Two qubit examples:

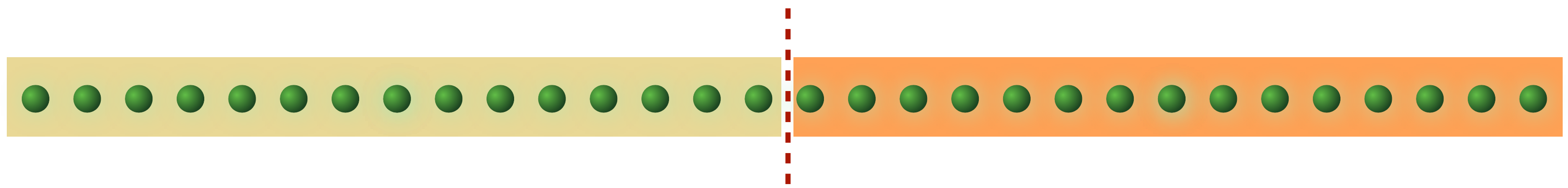
$$|\psi\rangle = |0\rangle_A |1\rangle_B \qquad \rho_A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}_A$$

$$S(\rho_A) = -1\log_2(1) = 0$$

$$|\psi\rangle = \frac{1}{2}\left(|0\rangle_A |0\rangle_B + |1\rangle_A |1\rangle_B\right) \qquad \rho_A = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}_A$$

$$S(\rho_A) = -2\frac{1}{2}\log_2\left(\frac{1}{2}\right) = \log_2(2) = 1$$

Lecture 3 - Singular value (Schmidt) decomposition



$= |\psi\rangle$

$d_A = d_B = d$

- General state in some basis: $|\psi\rangle = \sum_{i,j=1}^{d} c_{i,j} |i\rangle_A |j\rangle_B$ $\quad (\mathbf{C})_{i,j} = c_{i,j}$
- Now let's do a **singular value decomposition (SVD)** of the coefficient matrix:

$$c_{i,j} = \sum_{\alpha}^{d} u_{i,\alpha} s_{\alpha,\alpha} v^*_{\alpha,j} \qquad \mathbf{C} = \mathbf{U}\mathbf{S}\mathbf{V}^\dagger$$

U,**V** unitary, **S**: real & diagonal

"Singular values"

… linear algebra textbook

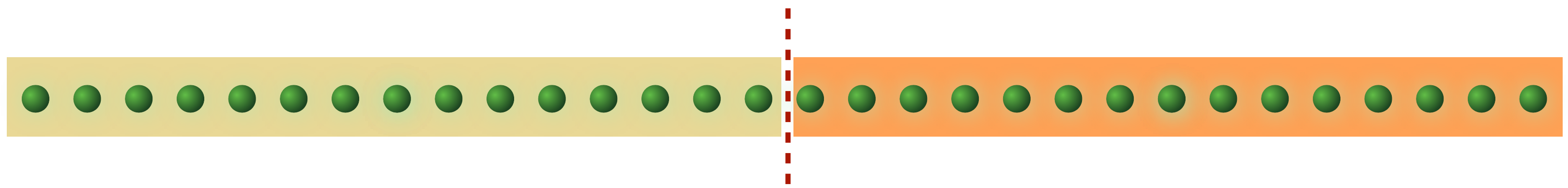
- Then it follows:

$$\rho_A = \mathrm{tr}_B \left(|\psi\rangle\langle\psi| \right) = \cdots = \sum_{\alpha} (s_{\alpha,\alpha})^2 |\alpha\rangle\langle\alpha|_A \qquad |\alpha\rangle_A = \sum_i u^*_{i,\alpha} |i\rangle_A$$

- The **squared singular values** (known as Schmidt values) are the **eigenvalues** of ρ_A !
- Entanglement entropy:

$$S_{\mathrm{vN}}(\rho_A) = -\sum_{\alpha}^{d} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2]$$

Lecture 3 - Truncating Entanglement!



$= |\psi\rangle$

$d_A = d_B = d$

$$|\psi\rangle = \sum_{i,j=1}^{d} c_{i,j} |i\rangle_A |j\rangle_B$$

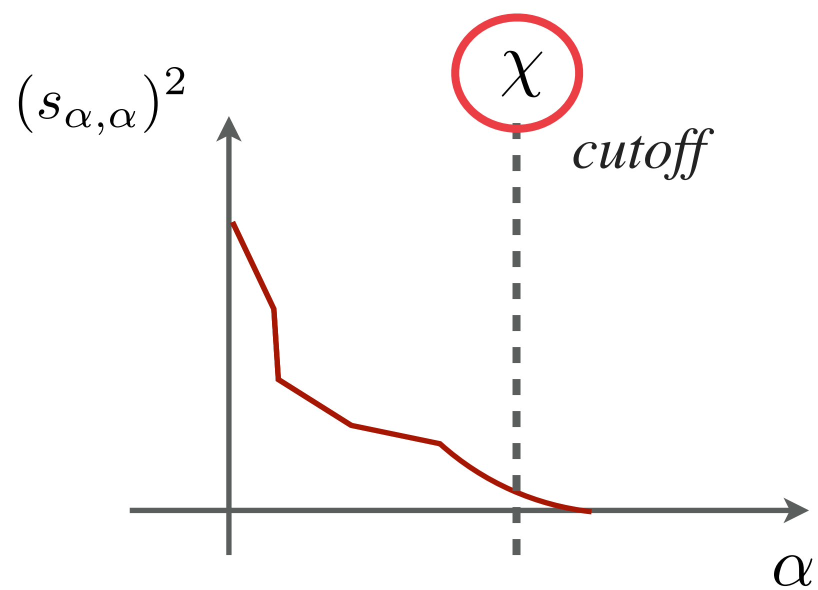
SVD

$$c_{i,j} = \sum_{\alpha}^{d} u_{i,\alpha} s_{\alpha,\alpha} v^*_{\alpha,j}$$

$$S_{\mathrm{vN}}(\rho_A) = -\sum_{\alpha}^{d} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2] \leq \log_2(d)$$

- Depending on state:

The singular values will follow some distribution!



Key idea!

Let's truncate small singular values/basis states

- **Then:** $S_{\mathrm{vN}}(\rho_A) = -\sum_{\alpha}^{d} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2] \approx -\sum_{\alpha}^{\chi} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2] \leq \log_2(\chi)$

- error: $= \sum_{\alpha=\chi+1}^{d} (s_{\alpha,\alpha})^2$ $\qquad \sum_{\alpha=1}^{d} (s_{\alpha,\alpha})^2 = 1 \approx \sum_{\alpha=1}^{\chi} (s_{\alpha,\alpha})^2$

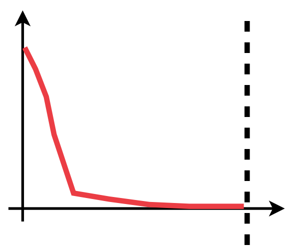
Lecture 3 - Truncating Entanglement!

- Example "chicken state": $|\psi\rangle = \sum_{i,j=1}^{d} c_{i,j} |i\rangle_A |j\rangle_B \qquad (\mathbf{C})_{i,j} = c_{i,j}$

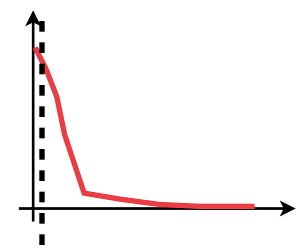
Matrix of gray-scale values of an image of a cat

$$\mathbf{C} = \mathbf{USV}^\dagger$$

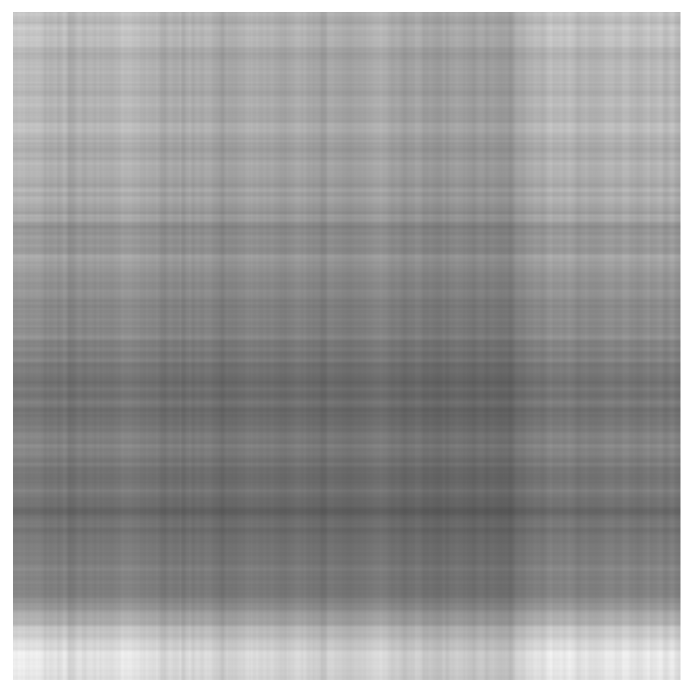
- Do SVD, then truncate, then re-multiply … one gets:



$\chi = d$

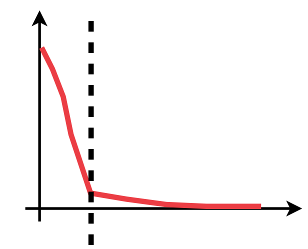


$\chi = 1$



mean-field chicken

$S_{\mathrm{vN}}(\rho_A) = 0$



$\chi = 0.05d$

slightly entangled chicken

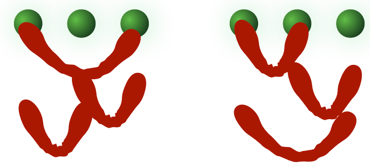
$S_{\mathrm{vN}}(\rho_A) \leq \log(\chi)$

- **Think of it:** truncated SVD = Image compression for quantum states in bipartitions.

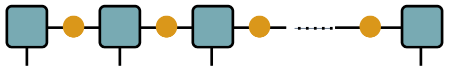
Plan for today

A strategy to simulate dynamics in systems way beyond 30 qubits - Matrix Product States

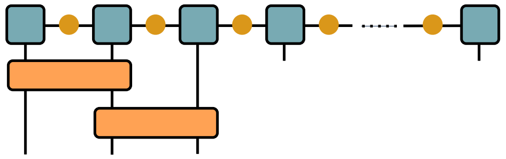
- Boring preliminaries: Suzuki-Trotter decomposition



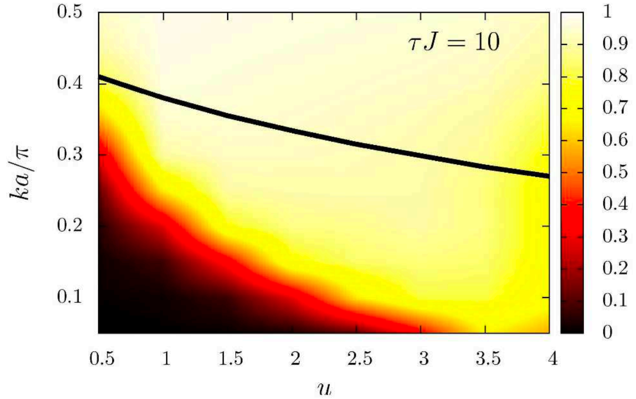
- Bipartite entanglement and singular value decompositions
- Matrix product states (MPS)



- The time-evolving block decimation algorithm (TEBD)

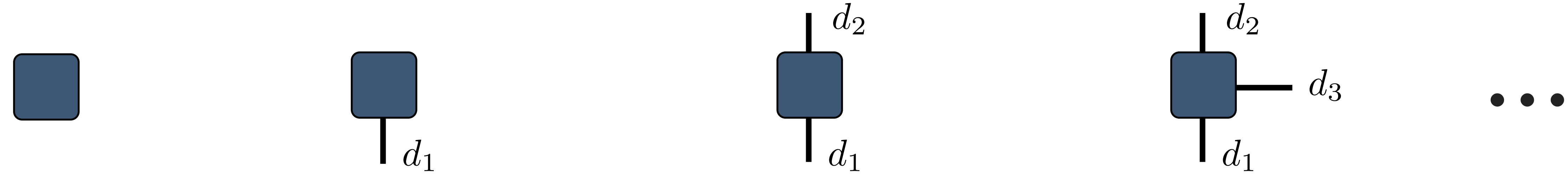


- Application example of Bose-Hubbard and Spin-model dynamics



Lecture 3 - Matrix product states (MPS)

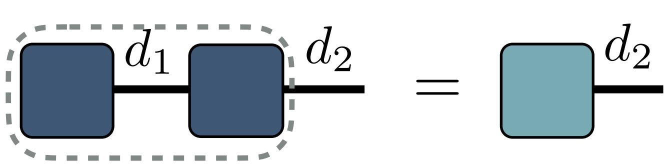
- **Introducing:** Diagrammatic tensor notation



Complex number	*Vector*	*Matrix*	*3D tensor*
z	v_i	M_{ij}	$\Gamma_{i,j,k}$
	$i = 1, \ldots, d_1$	$i = 1, \ldots, d_1$ $j = 1, \ldots, d_2$	$i = 1, \ldots, d_1$ $j = 1, \ldots, d_2$ $k = 1, \ldots, d_3$

- Useful for avoiding messy indexing in tensor operations, allows easy estimations of numerical complexity:

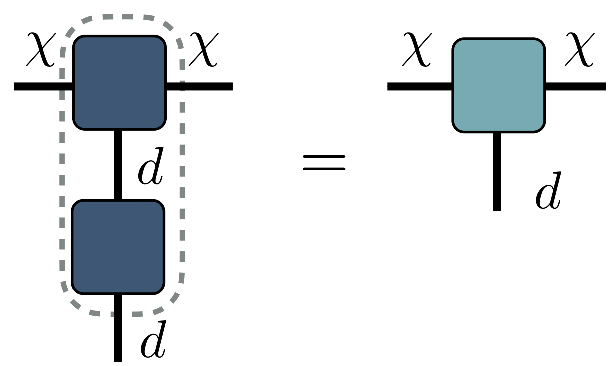
Matrix-Vector product:



$$\sum_{i=1}^{d_1} v_i M_{ij} = \tilde{v}_j$$

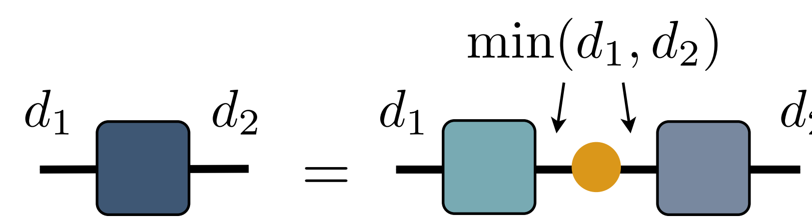
$\mathcal{O}(d_1 d_2)$ *operations*

Tensor-matrix contraction:



$\mathcal{O}(d^2\chi^2)$ *operations*

SVD:



$$\mathbf{C} = \mathbf{USV}^\dagger$$

Depending on algorithm, usually

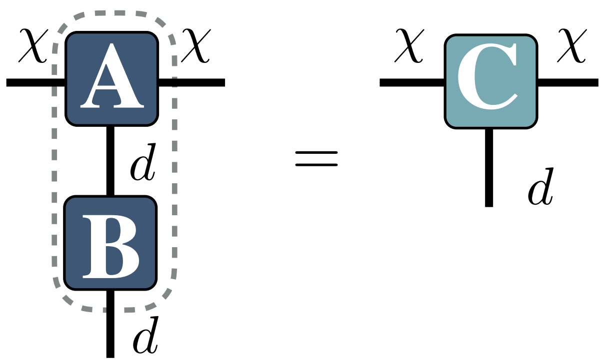
$$\mathcal{O}(\min(d_1^2 d_2, d_1 d_2^2))$$

- Many useful tensor libraries available, e.g. https://itensor.org/ (Julia/C++)

Lecture 3 - Matrix product states (MPS)

- Many useful tensor libraries available, e.g. https://itensor.org/ (Julia/C++)

Tensor-matrix contraction:



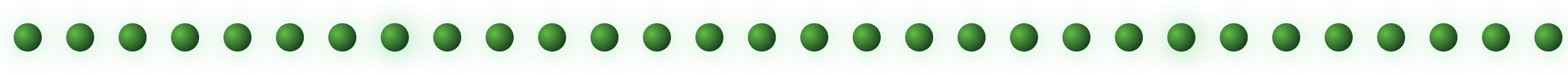
$\mathcal{O}(d^2\chi^2)$ *operations*

```
using ITensors

chi = 512
d = 2

# define indices
Cleft = Index(chi, "Cleft")       # left horizontal index
Cright = Index(chi, "Cright")     # right horizontal index
iup = Index(d, "iupper")          # upper vertical index
ilo = Index(d, "ilower")          # lower vertical index

A = randomITensor(Cleft, iup, Cright)
B = randomITensor(ilo, iup)

C = A*B
```

```
julia> C = A*B
ITensor ord=3 (dim=512|id=31|"Cleft") (dim=512|id=112|"Cright") (dim=2|id=521|"ilower")
```

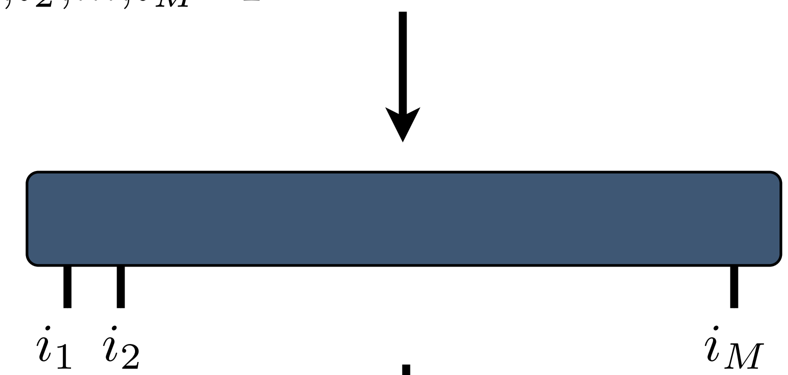
Lecture 3 - Matrix product states (MPS)

- Many-body quantum state (local dimension *d*):

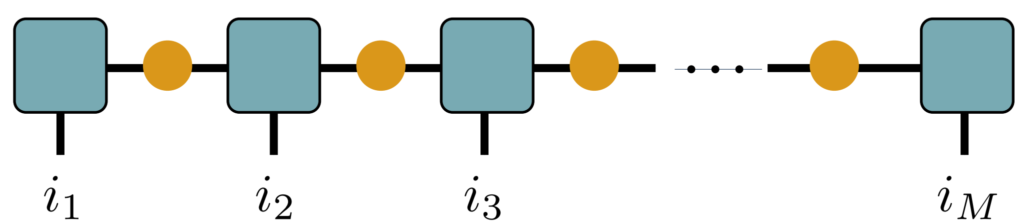


$= |\psi\rangle$

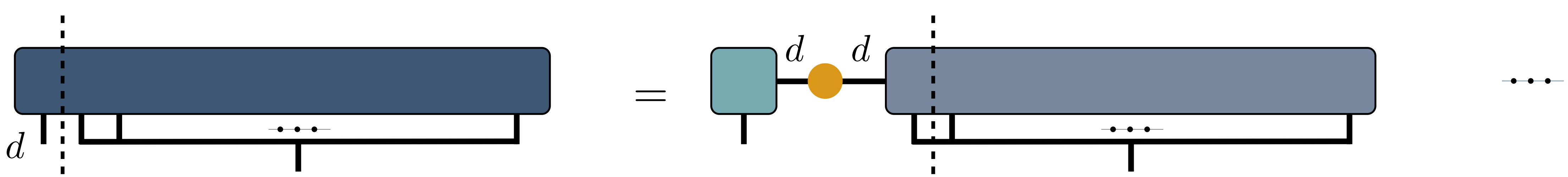
$$|\psi\rangle = \sum_{i_1,i_2,\ldots,i_M=1}^{d} c_{i_1,i_2,\ldots,i_M} |i_1, i_2, \cdots {}_M\rangle$$



- **Matrix product state:**



- … can be done iteratively with SVDs



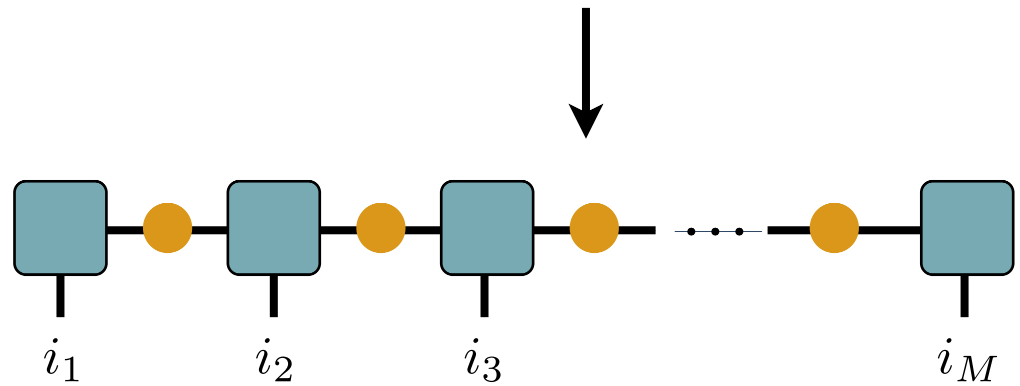
combine indices 2,…,M

iterate this

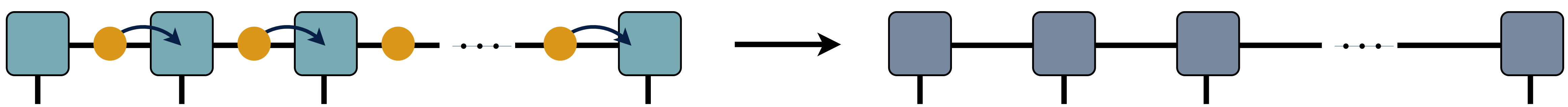
Lecture 3 - Matrix product states (MPS)

$$|\psi\rangle = \sum_{i_1,i_2,\ldots,i_M=1}^{d} c_{i_1,i_2,\ldots,i_M} |i_1, i_2, \cdots_M\rangle$$

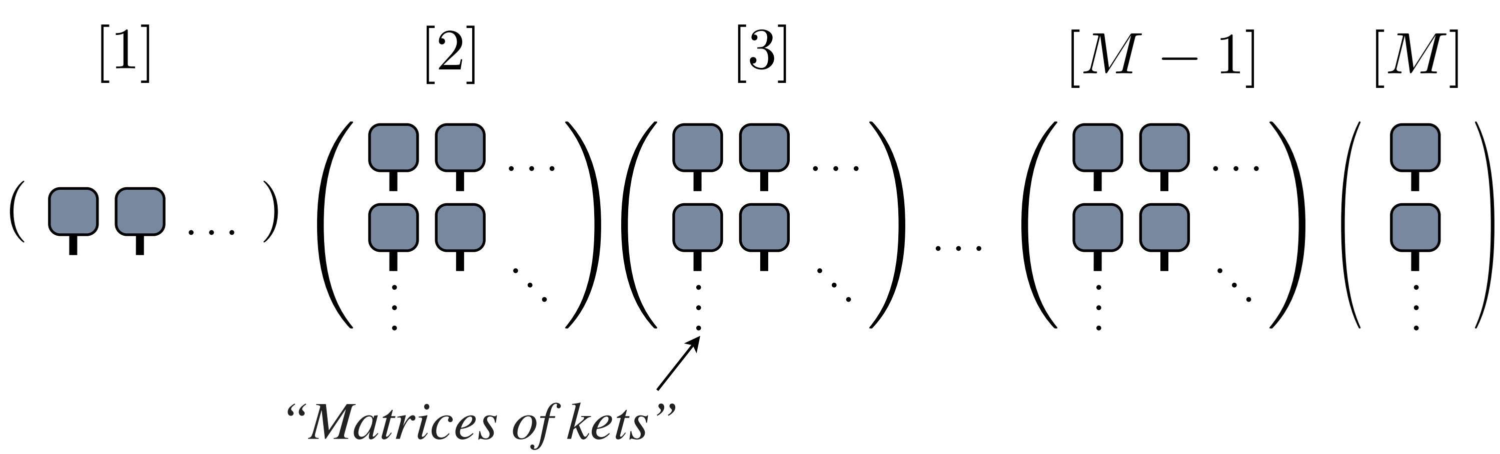
- **Matrix product state:**



- **Note 1:** Diagonal singular value matrices can be multiplied into other tensors



- **Note 2:** Another way to think about this decomposition:



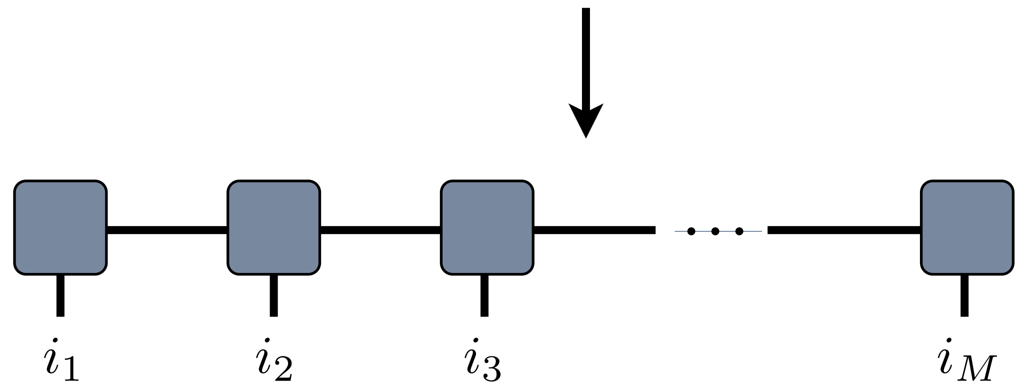
Matrix product

Hence the name: **Matrix product state**

Lecture 3 - Matrix product states (MPS)

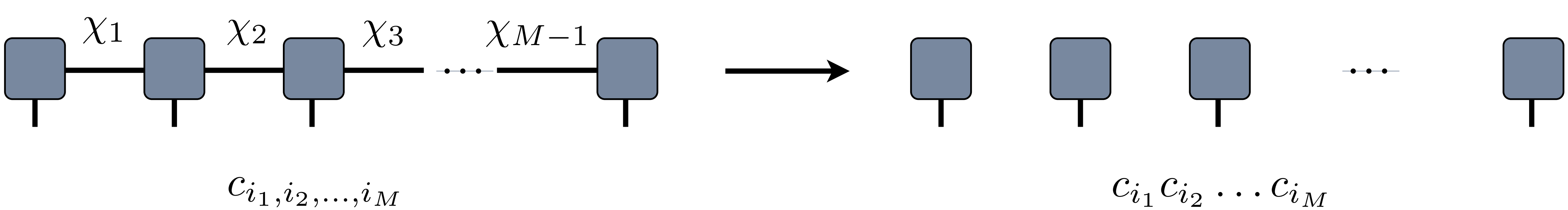
$$|\psi\rangle = \sum_{i_1, i_2, \ldots, i_M = 1}^{d} c_{i_1, i_2, \ldots, i_M} |i_1, i_2, \cdots_M\rangle$$

- **Matrix product state:**



- **Note 3:** A product state is a special case of a matrix product state:

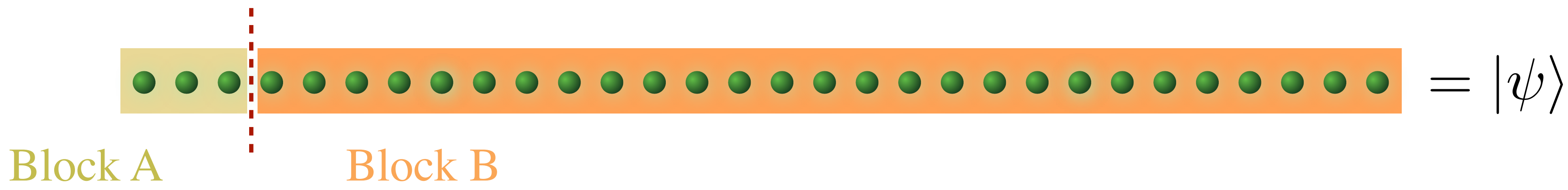
If all horizontal dimensions are: $\chi_i = 1$

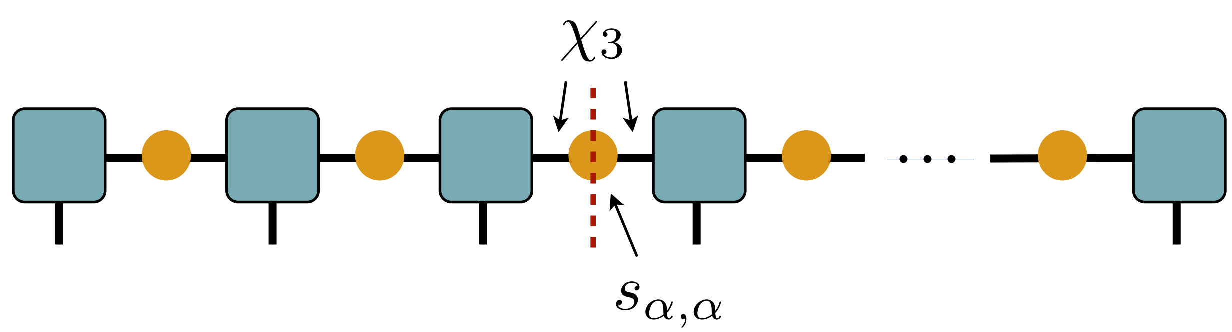


No entanglement!

Lecture 3 - Matrix product states (MPS)

- **Note 4 (crucial):**





The horizontal dimension determine the amount of bipartite entanglement that can be stored!

- Since the MPS is constructed from consecutive SVDs

Block A
1, 2, …, i

Bipartite entanglement entropy

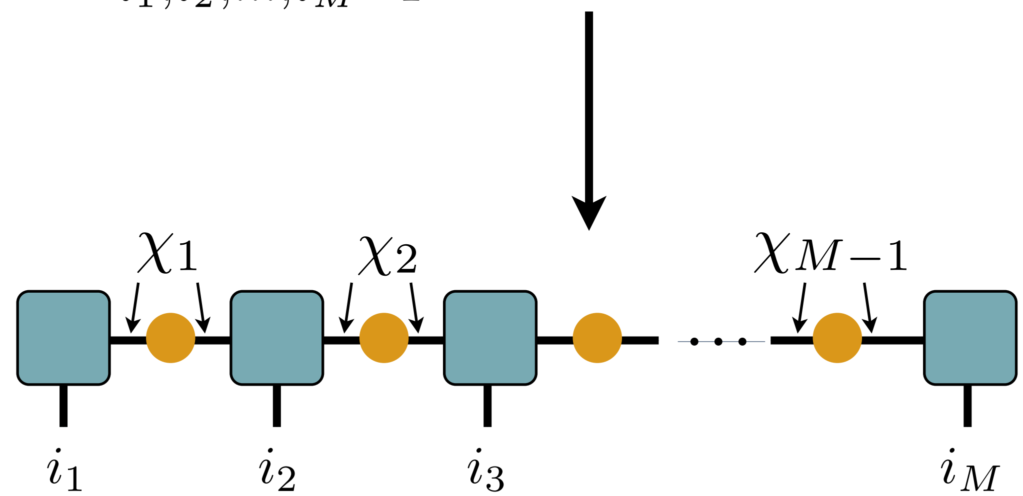
Block B
i+1, i+2, …, M

$$S_{\mathrm{vN}}(\rho_A^{[i]}) = -\sum_{\alpha}^{\chi_{i+1}} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2] \leq \log_2(\chi_{i+1})$$

Note: $\chi_{i+1} = 1$ → $S_{\mathrm{vN}}(\rho_A^{[i]}) = 0$ → *product state*

Lecture 3 - Matrix product states (MPS)

$$|\psi\rangle = \sum_{i_1,i_2,\ldots,i_M=1}^{d} c_{i_1,i_2,\ldots,i_M} |i_1, i_2, \cdots_M\rangle$$



$$S_{\mathrm{vN}}(\rho_A^{[i]}) = -\sum_{\alpha}^{\chi_{i+1}} (s_{\alpha,\alpha})^2 \log_2[(s_{\alpha,\alpha})^2] \leq \log_2(\chi_{i+1})$$

- In order for the MPS representation to be exact:

$\chi_1 = d$ | $\chi_2 = d^2$ | $\cdots$ | $\chi_{\frac{M}{2}} = d^{\frac{M}{2}}$ | $\cdots$ | $\chi_{M-2} = d^2$ | $\chi_{M-1} = d$

This is still growing exponentially with M

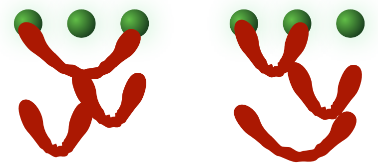
- **But:** **Key idea - Introduce a finite cut-off** $\chi_i \leq \chi$
- **Then:** An MPS is a state decomposition that controls the amount of entanglement in a state!

An MPS with "bond dimension" χ ⟷ *Storage space* $\mathcal{O}(M\chi^2 d)$ ⟷ $S_{\mathrm{vN}}(\rho_A^{[i]}) \leq \log_2(\chi)$

Plan for today

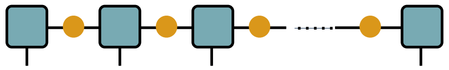
A strategy to simulate dynamics in systems way beyond 30 qubits - Matrix Product States

- Boring preliminaries: Suzuki-Trotter decomposition

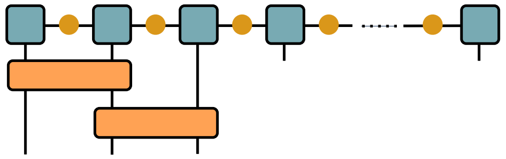


- Bipartite entanglement and singular value decompositions

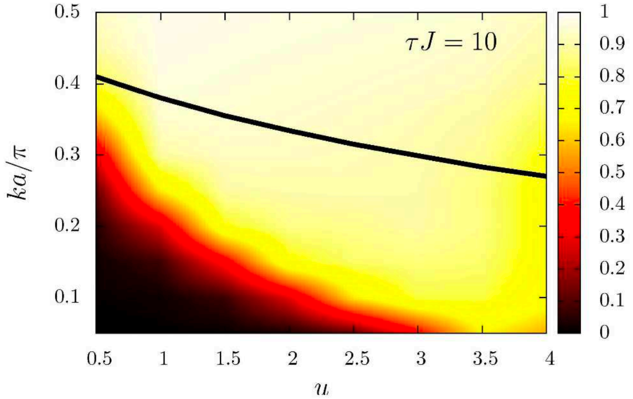
- Matrix product states (MPS)



- The time-evolving block decimation algorithm (TEBD)

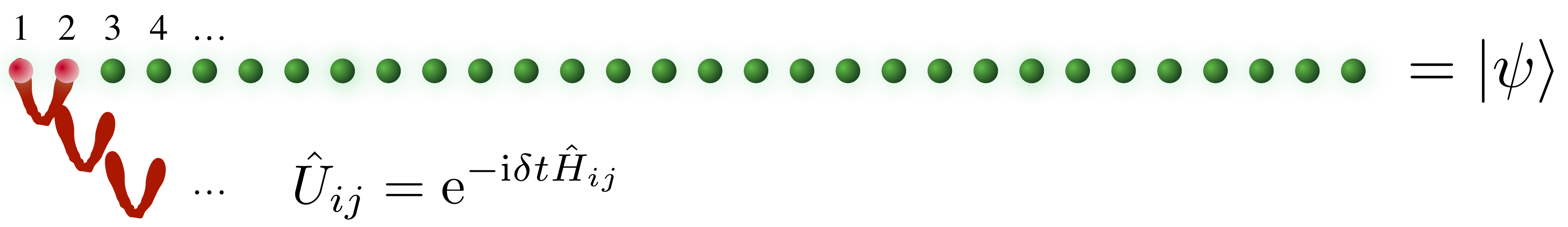


- Application example of Bose-Hubbard and Spin-model dynamics

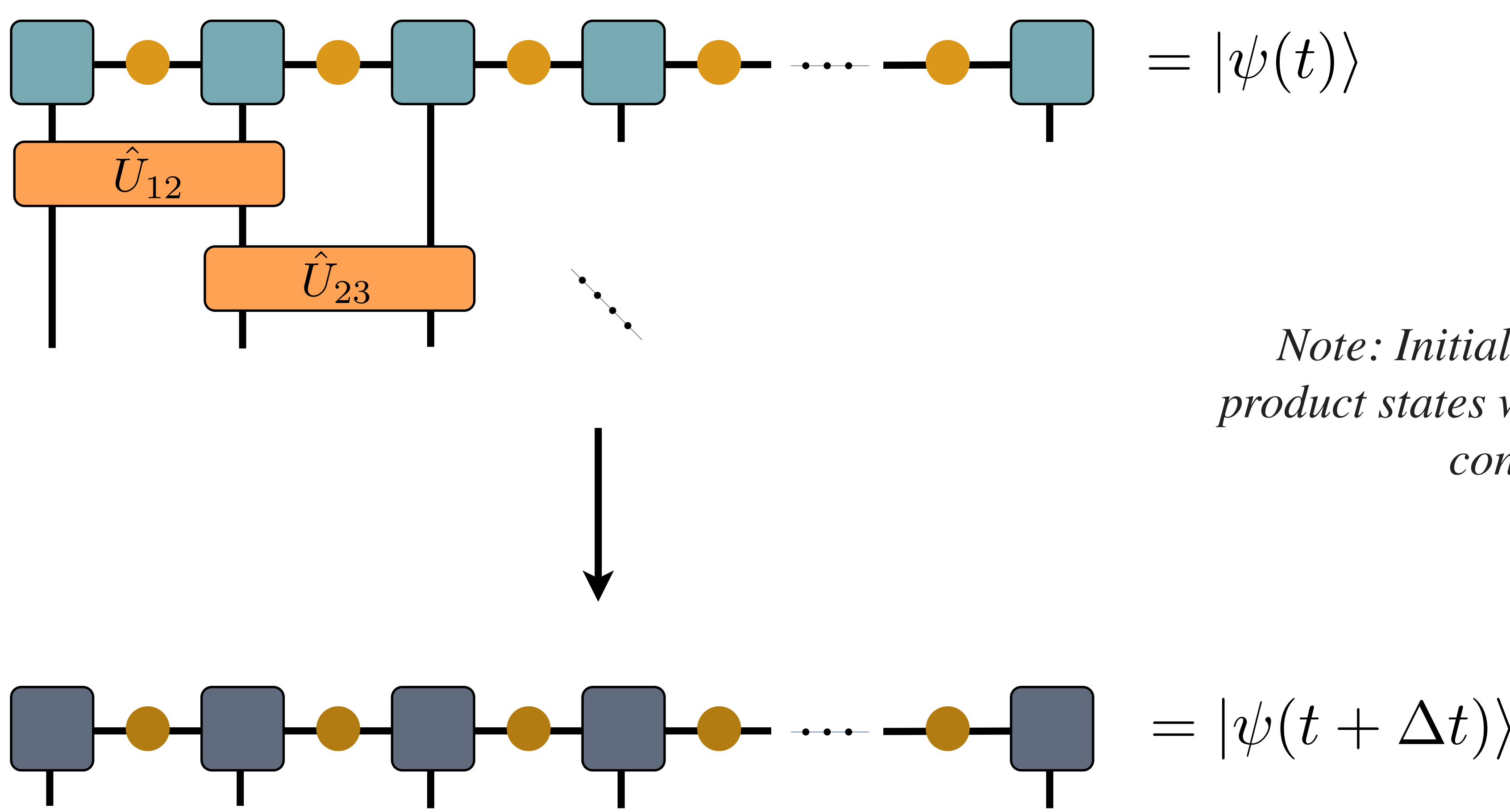


Lecture 3 - The TEBD algorithm … bringing it all together:

- **1.** A many-body time-evolution over a timestep can be written as application of two-site unitaries: (Trotter decomposition, nearest neighbors only w.l.o.g.)



- **2.** In the matrix product state language … we therefore need to update the MPS like:



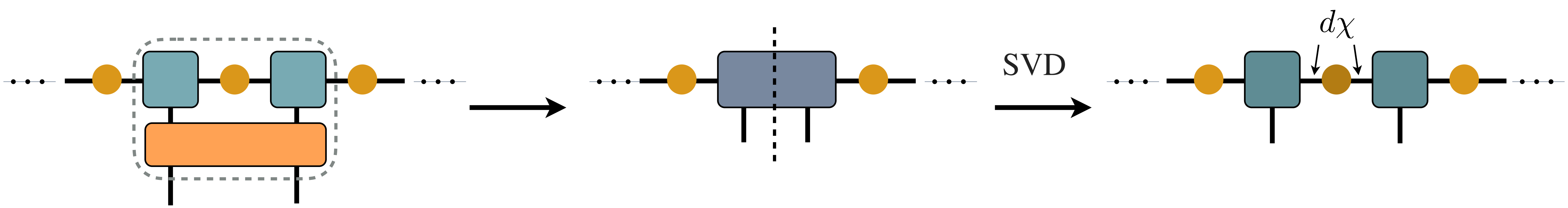
Note: Initial states are usually product states which can be trivially constructed.

Lecture 3 - The TEBD algorithm … bringing it all together:

- **3.** Two-site updates of the MPS can be done with the TEBD algorithm (time-evolving block decimation)

G. Vidal, Phys. Rev. Lett. 93, 040502 (2004)

… which says:



$\mathcal{O}(\chi^3 d^2)$

The new horizontal dimension can grow = *The gate builds up entanglement*

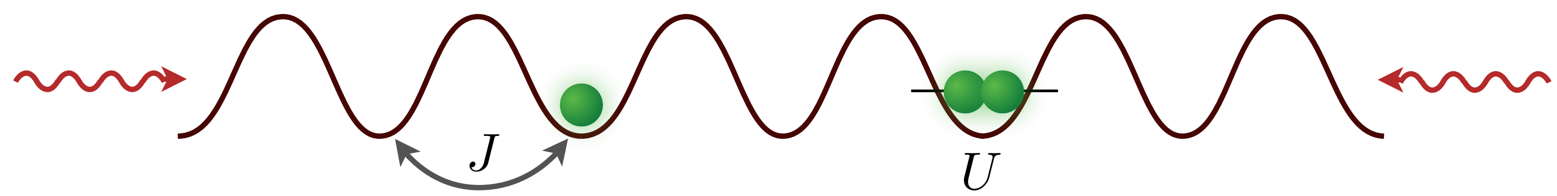
Truncation needed! Error

$$\epsilon = \sum_{\alpha=\chi+1}^{d\chi} (s_{\alpha,\alpha})^2$$

- **4.** If the dynamics does not build up much entanglement between the *M-1* bipartitions, the error stays small and the simulation is exact!

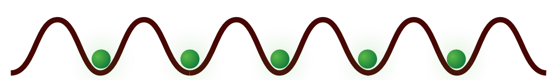
Lecture 3 - Bose-Hubbard evolution

- **Model:** Bosonic atoms trapped in an optical lattice



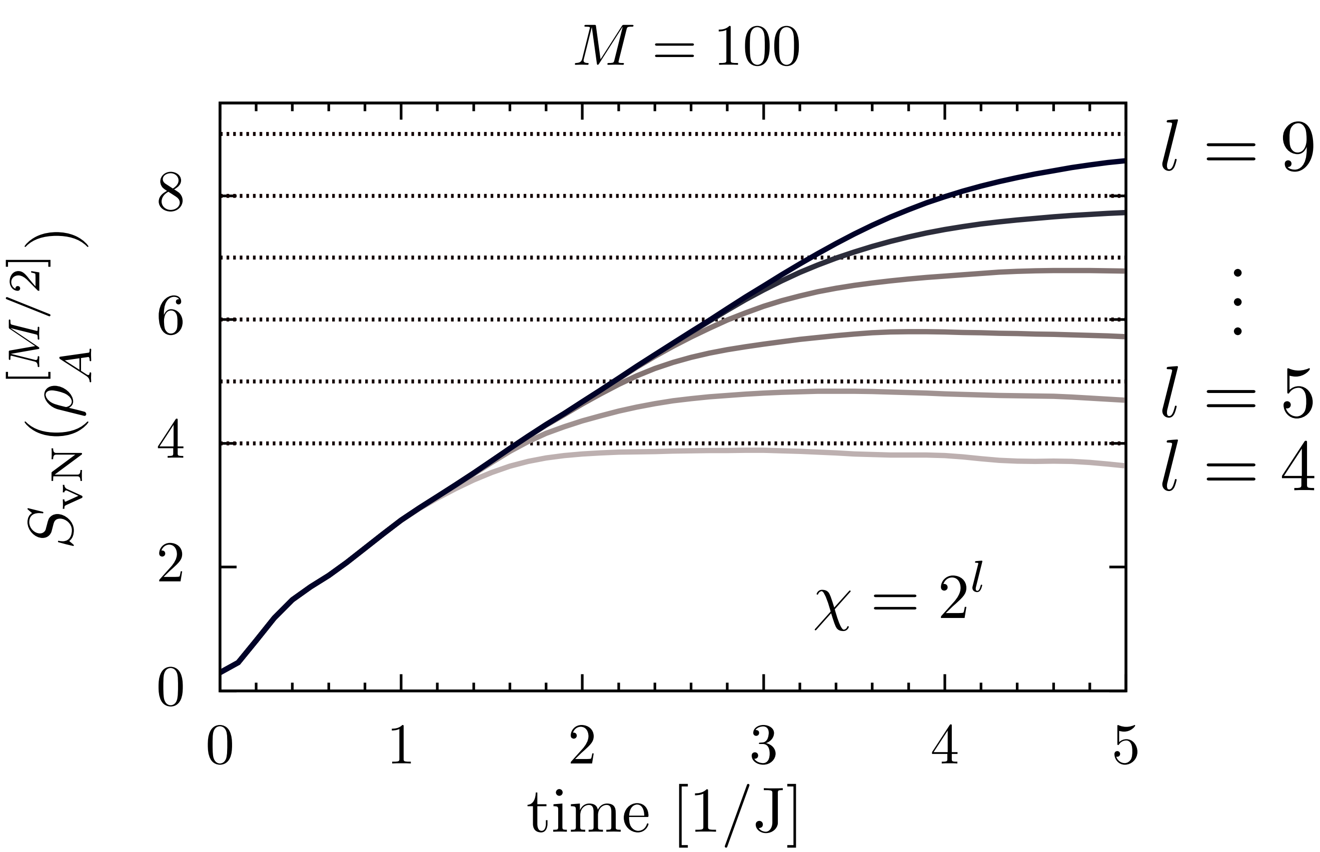
$$\hat{H} = -J \sum_i (\hat{b}_i \hat{b}_{i+1}^\dagger + \hat{b}_i^\dagger \hat{b}_{i+1}) + \frac{U}{2} \sum_i \hat{n}_i(\hat{n}_i - 1) \qquad \hat{n}_i = \hat{b}_i^\dagger \hat{b}_i$$

- **Experiment:** Initial state



$U \gg J \quad \rightarrow U \ll J$

sudden quench!



Linear growth

$S_{\mathrm{vN}}(t) \propto t$

$S_{\mathrm{vN}}(t) \propto \log(\chi)$

$\chi \propto 2^t$

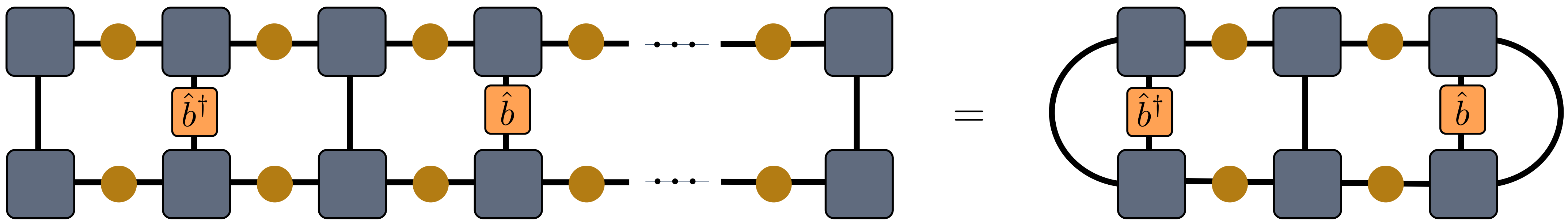
Hard to simulate long times, but several tunneling times feasible

from A. J. Daley, H. Pichler, JS, P. Zoller, Phys. Rev. Lett. 109, 020505 (2012)

Lecture 3 - Observables and ground states

- **Remark 1:** Observables can be easily numerically calculated.

Single particle density matrix element $\langle \hat{b}_2^\dagger b_4 \rangle$



- **Remark 2:** What if we're interested in computing ground states?

Imaginary time-evolution

$$|\psi(\tau)\rangle = \mathrm{e}^{-\tau \hat{H}} |\psi(t=0)\rangle = \sum_n \mathrm{e}^{-\tau E_n} |\psi_n\rangle \langle \psi_n | \psi(t=0)\rangle = \sum_n c_n(\tau) |\psi_n\rangle$$

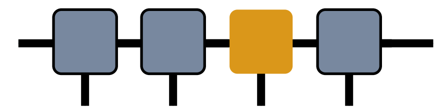
$$\hat{H} = \sum_n E_n |\psi_n\rangle \langle \psi_n| \qquad E_0 < E_1 < E_2 < \dots$$

Then, at late times: $c_0(\tau \to \infty) \gg c_1(\tau \to \infty) \gg c_2(\tau \to \infty) \dots$

When normalizing: $\tilde{c}_i(\tau) = \dfrac{c_i(\tau)}{\sum_n c_n(\tau)^2}$ $\quad \tilde{c}_i(\tau \to \infty) = \delta_{i,0}$ *Only the GS survives!*

Lecture 3 - Further reading … we were just scratching the surface

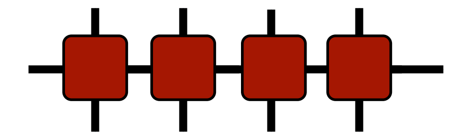
- Variational techniques
  (e.g. ground-state finding = **DMRG**)



e.g. local updates to minimize energy

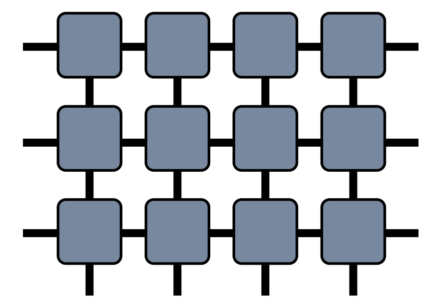
"density matrix renormalization group" *S. White, 1992*

- Long range interactions: Matrix product operators (MPO) of **Hamiltonians**

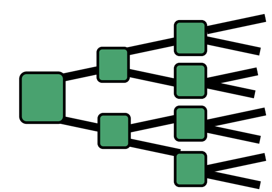
$\hat{H} =$ 

"projected entangled pair states"

- Elegant formulations for higher dimensional models (PEPS)

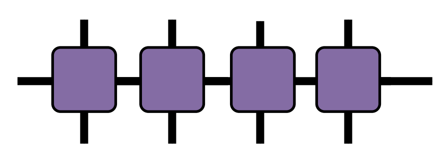
2D 

- Multi-scale entanglement renormalization ansatz (MERA):

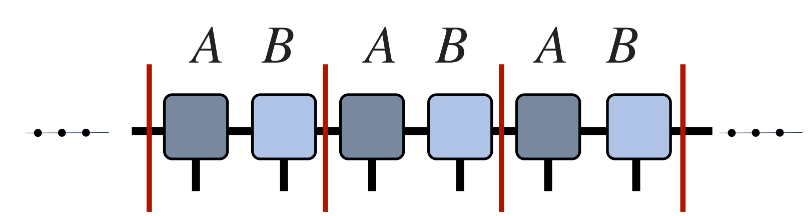


"Tensor network trees"

- Open systems: Density matrix as MPO

$\rho =$ 

- Infinite systems (iTEBD) … exploiting translational symmetries:



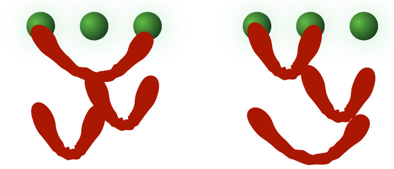
- **Further reading:** Review article with references:

U. Schollwöck, Ann. Phys. 326, 96 (2011), arXiv:1008.3477

Plan for today

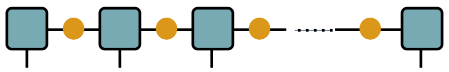
A strategy to simulate dynamics in systems way beyond 30 qubits - Matrix Product States

- Boring preliminaries: Suzuki-Trotter decomposition

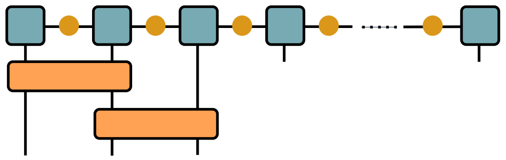


- Bipartite entanglement and singular value decompositions

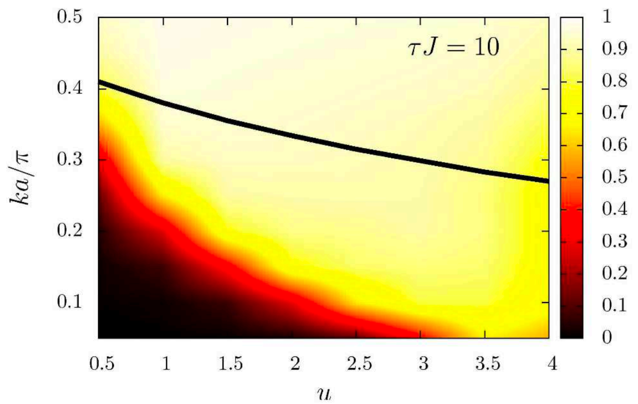
- Matrix product states (MPS)



- The time-evolving block decimation algorithm (TEBD)

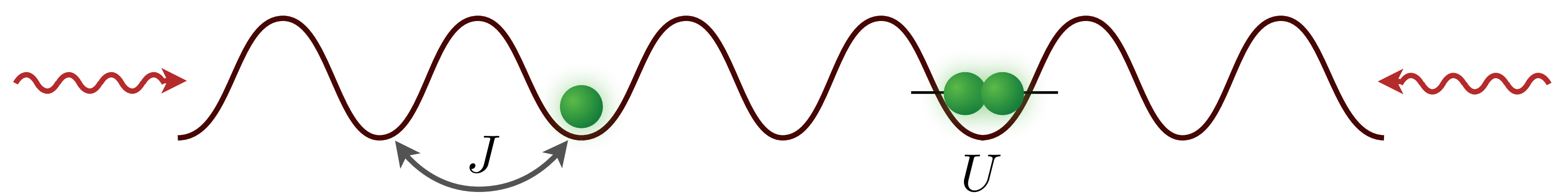


- Application example of Bose-Hubbard and Spin-model dynamics



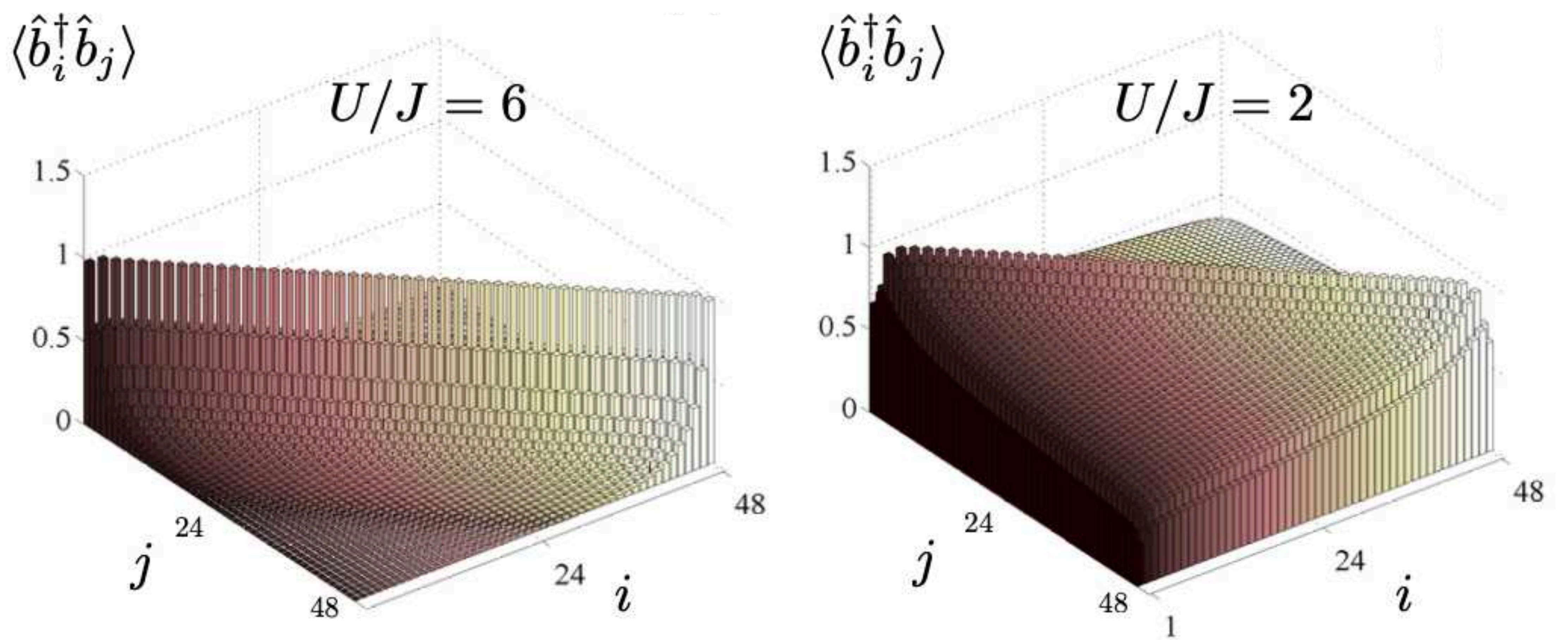
Lecture 3 - Ground state phase transition in Bose-Hubbard models:

- **Model:** Bosonic atoms trapped in an optical lattice



$$\hat{H} = -J \sum_i (\hat{b}_i \hat{b}_{i+1}^\dagger + \hat{b}_i^\dagger \hat{b}_{i+1}) + \frac{U}{2} \sum_i \hat{n}_i(\hat{n}_i - 1) \qquad \hat{n}_i = \hat{b}_i^\dagger \hat{b}_i$$

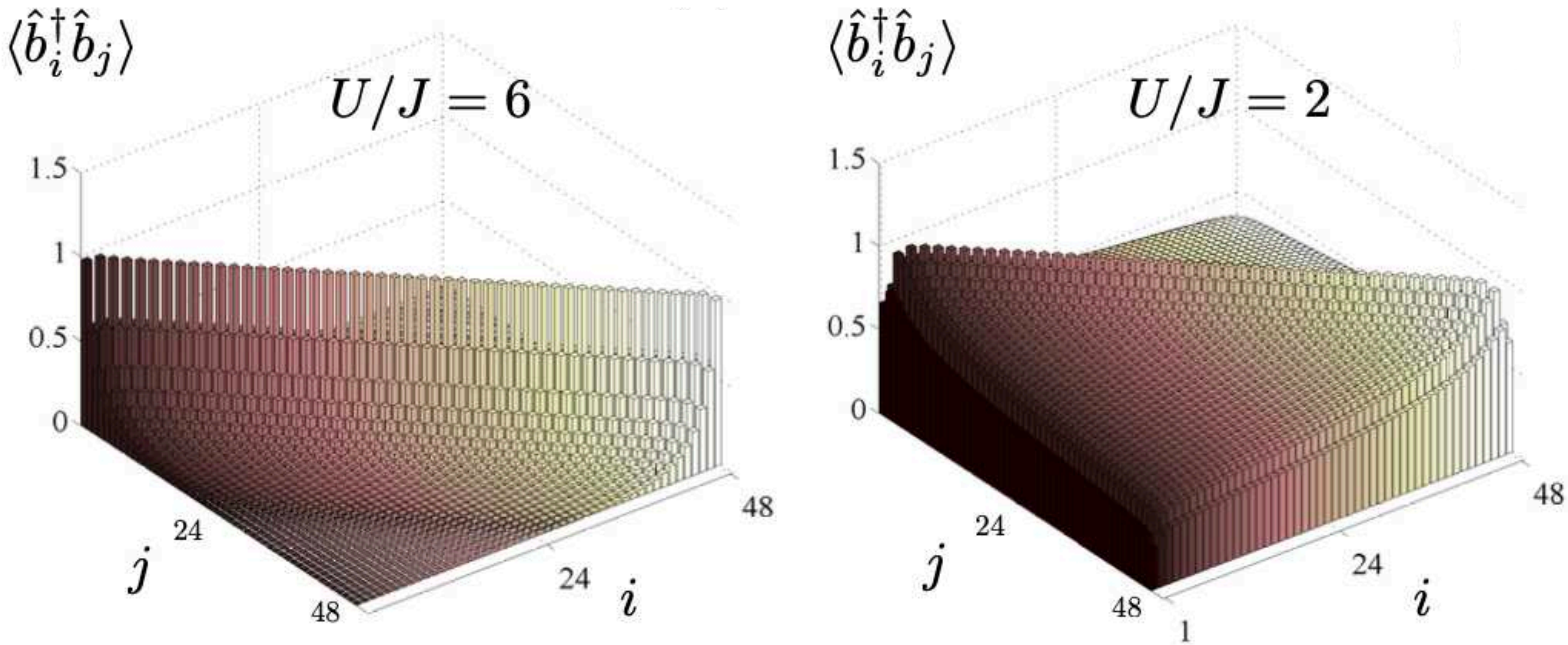
- **Model:** Exact MPS calculation of single-particle density matrix for *N=48* bosons on *M=48* sites:



Lecture 3 - Ground state phase transition in Bose-Hubbard models:

- **Model:** Exact MPS calculation of single-particle density matrix for *N=48* bosons on *M=48* sites:

$$\hat{H} = -J\sum_i(\hat{b}_i\hat{b}_{i+1}^\dagger + \hat{b}_i^\dagger\hat{b}_{i+1}) + \frac{U}{2}\sum_i \hat{n}_i(\hat{n}_i - 1)$$



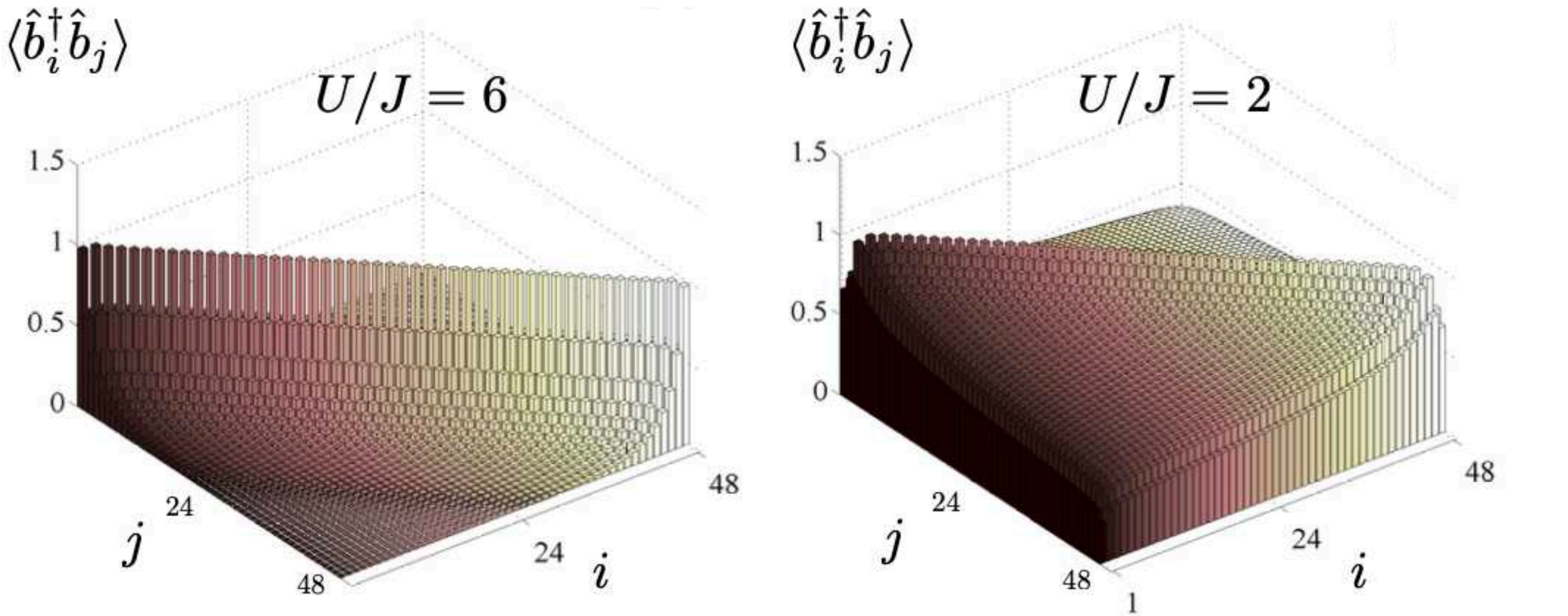
- **Observation:** There is **no** off-diagonal long-range order. This is a peculiarity of one dimension!
- **However:**

For strong interactions, the SPDM show exponential off-diagonal decay, for weak interactions it shows only a power-law decay. The latter is known as **quasi off-diagonal long-range order**!

Lecture 3 - Ground state phase transition in Bose-Hubbard models:

- **Model:** Exact MPS calculation of single-particle density matrix for *N=48* bosons on *M=48* sites:

$$\hat{H} = -J \sum_i (\hat{b}_i \hat{b}_{i+1}^\dagger + \hat{b}_i^\dagger \hat{b}_{i+1}) + \frac{U}{2} \sum_i \hat{n}_i (\hat{n}_i - 1)$$



This is a manifestation of a **quantum phase transition** (at *T=0*), here between a **Mott insulating phase** *(left, bosons pinned on sites)* and a **superfluid phase** *(bosons delocalized, quasi off-diagonal long range order)*

- **In 1D MPS calculations are really necessary**, transition point according to mean-field:

$(U/J)_C = 2(1 + \sqrt{2})2 \approx 11.66$ *Reality (MPS):* $(U/J)_C \approx 2.9$

Reminder - GP simulations with Runge-Kutta

$$\frac{d}{dt}\psi(x,t) = -\mathrm{i}\left(\frac{\hat{p}^2}{2m} + V(x) + g|\psi(x,t)|^2\right)\psi(x,t)$$

$$J = \frac{1}{2ma^2} \equiv 1$$

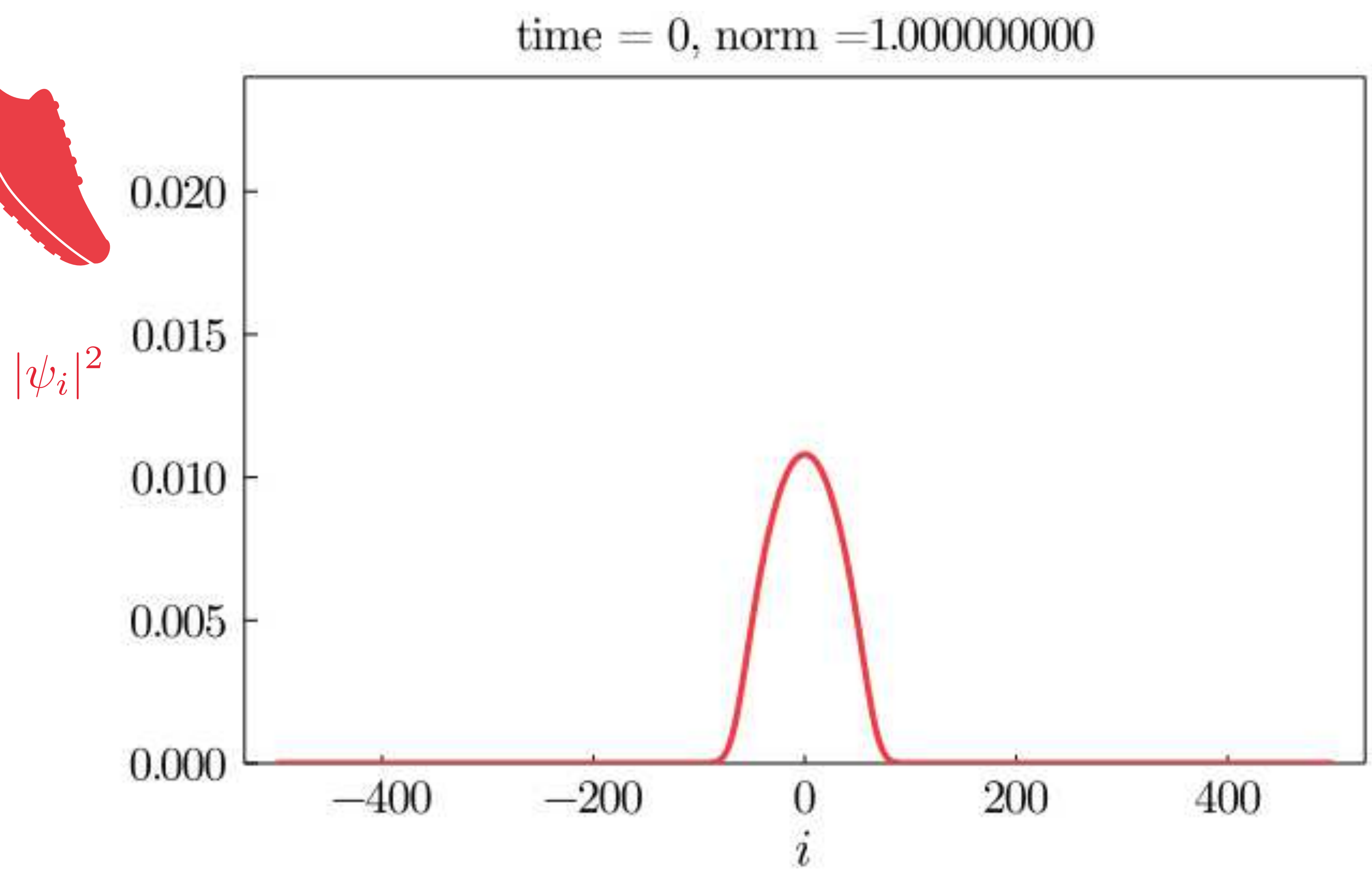
Solving it with RK4

$hJ = 0.02$

1001 grid points

periodic boundaries

- We kick it stronger
- … and kick the system! This can be done by applying a phase-gradient $\psi_i \rightarrow \psi_i e^{\mathrm{i}(ka)i}$



$g = 5J$

$ka = 0.4\pi$

Our BEC get's destroyed!

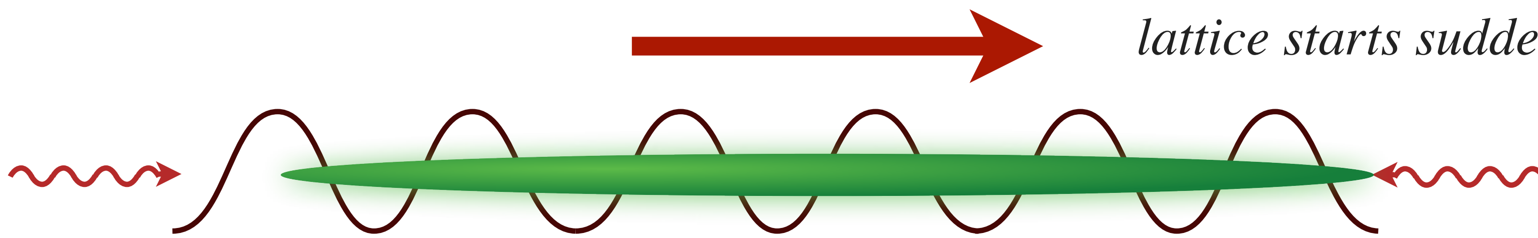
This is known as dynamical instability!

Lecture 3 - Dynamical instability of a superfluid

- **An experiment on stability of superfluid currents:**

J. Mun, P. Medley, G. K. Campbell, L. G. Marcassa, D. E. Pritchard, and W. Ketterle, Phys. Rev. Lett. 99, 150604 (2007)

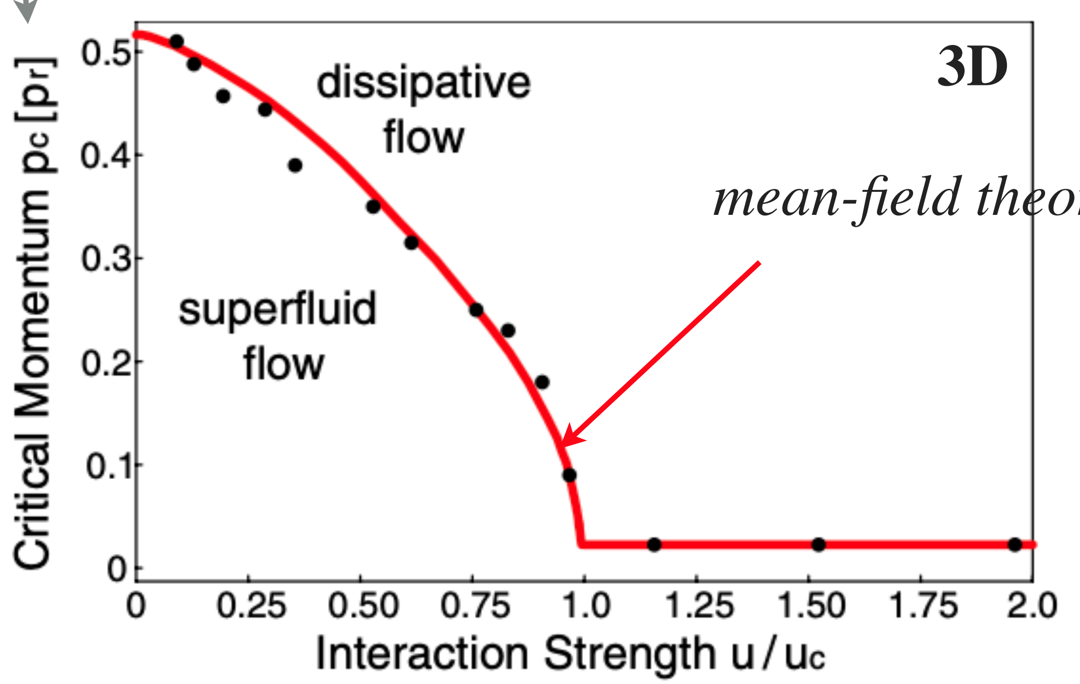
lattice starts suddenly moving



"momentum kick"

Measure fraction of particle in the condensate

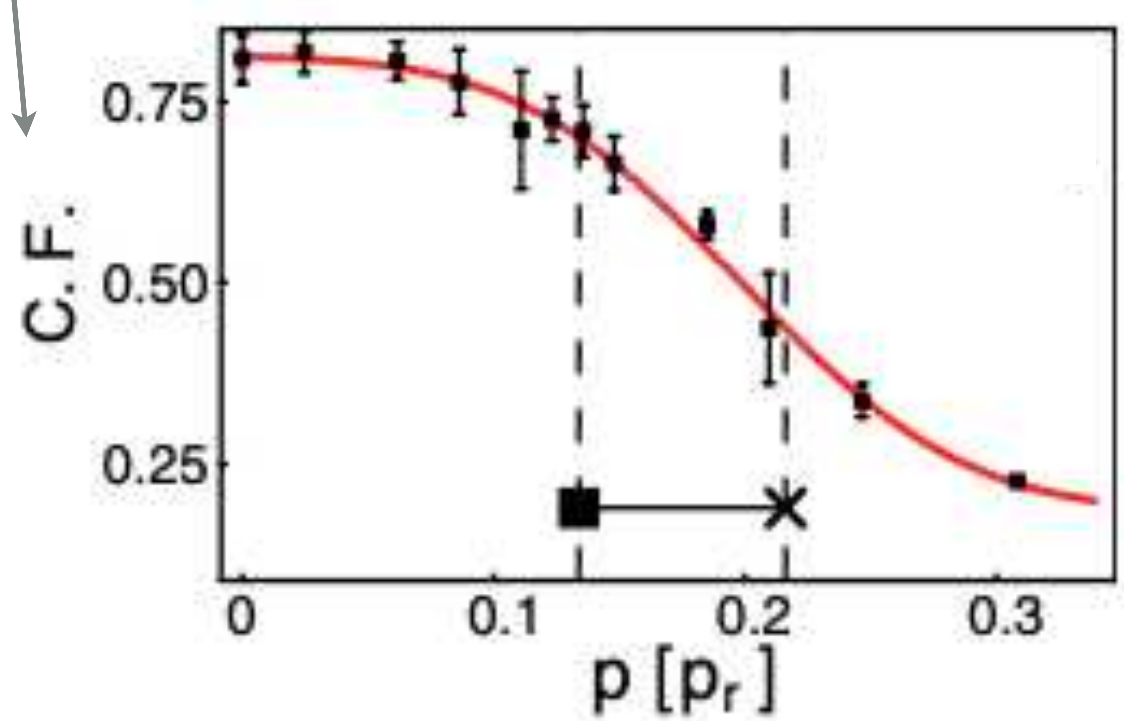
…amount of kick



mean-field theory

$u = U/J$

1D: mean-field failed badly

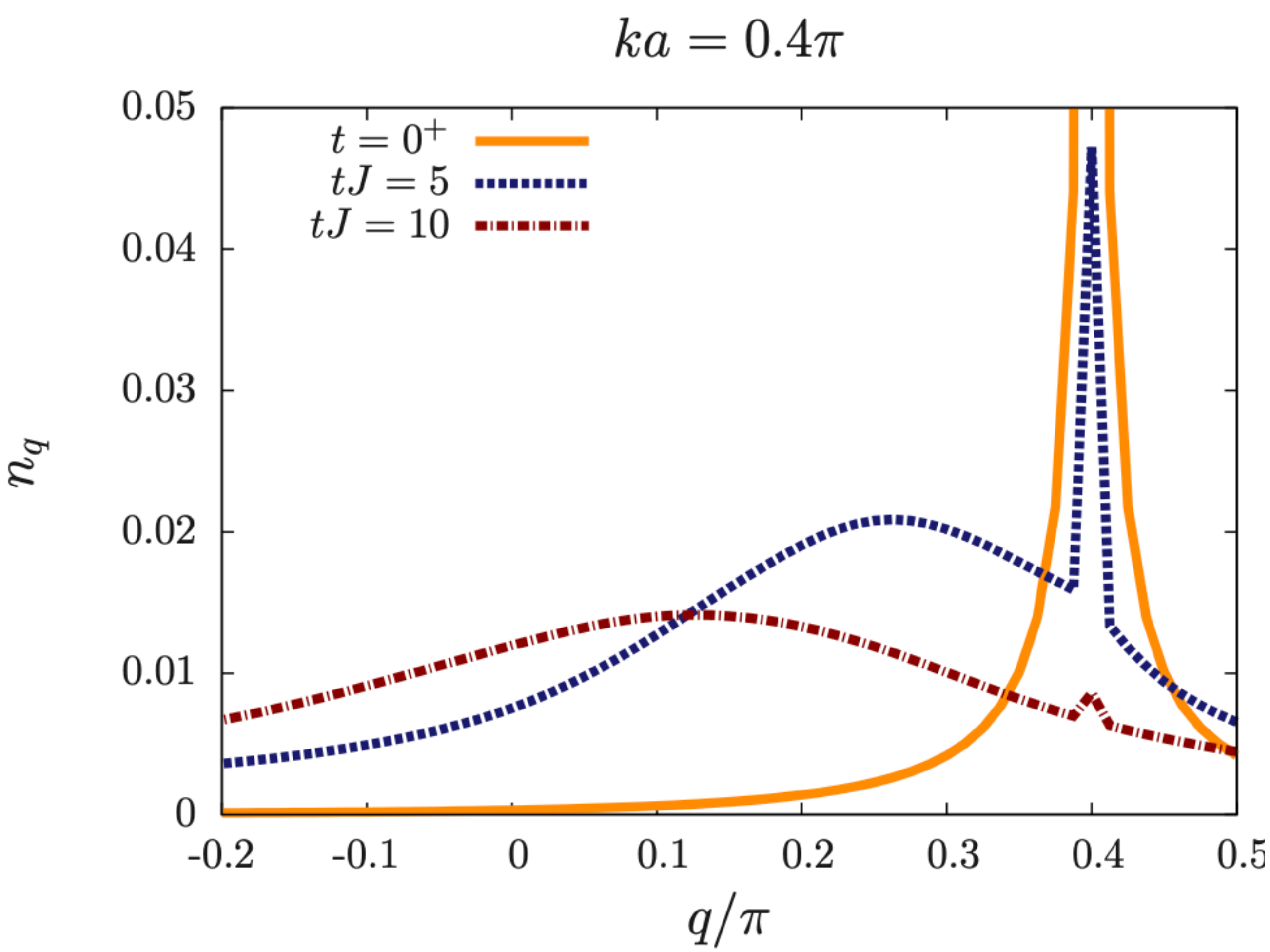


…no clear transition

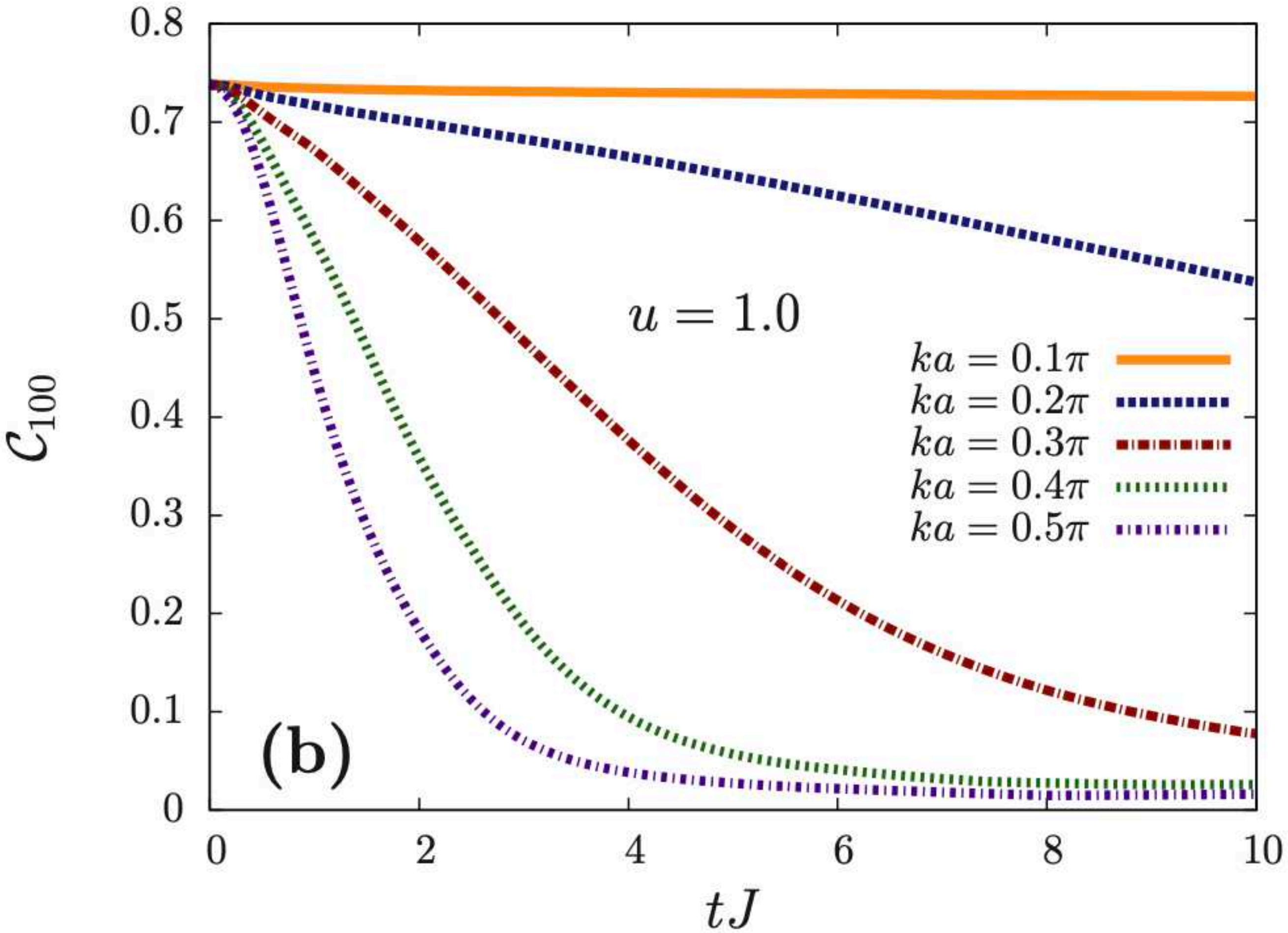
Lecture 3 - Dynamical instability of a superfluid

- **In 1D:** We can use MPS simulations

Momentum distributions



Condensate fraction

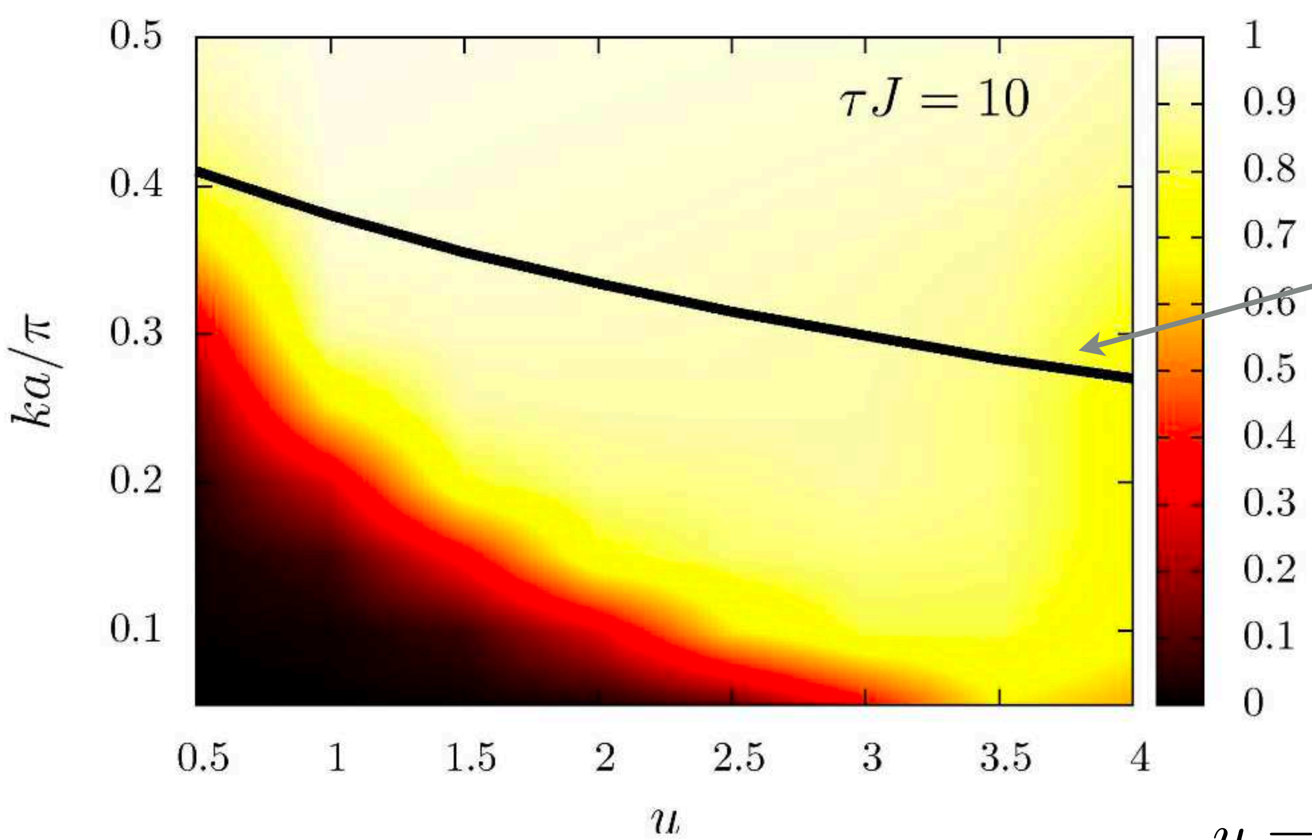


$u = U/J$

Lecture 3 - Dynamical instability of a superfluid

- We can use MPS to simulate the **loss in condensate function at a certain time**:

$$\Delta \mathcal{C}_R(\tau) \equiv |\mathcal{C}_R(t=\tau) - \mathcal{C}_R(t=0)|/\mathcal{C}_R(t=0)$$



mean-field does not work!

$u = U/J$

New J. Phys. 12, 025014 (2010)

Lecture 3 - Some recent results on entanglement dynamics in open spin-models

State vector $|\psi\rangle$ → *Density matrix* $\hat{\rho}$

Two strategies for **matrix product decomposition** of the density matrix:

Direct evolution of matrix product decomposition of density operator (MPDO):

$$\rho \approx \mathrm{tr} \begin{pmatrix} \hat{\rho}_1^{11} & \hat{\rho}_1^{12} & \\ \hat{\rho}_1^{21} & \hat{\rho}_1^{22} & \\ & & \ddots \end{pmatrix} \begin{pmatrix} \hat{\rho}_2^{11} & \hat{\rho}_2^{12} & \\ \hat{\rho}_2^{21} & \hat{\rho}_2^{22} & \\ & & \ddots \end{pmatrix} \cdots \begin{pmatrix} \hat{\rho}_N^{11} & \hat{\rho}_N^{12} & \\ \hat{\rho}_N^{21} & \hat{\rho}_N^{22} & \\ & & \ddots \end{pmatrix}$$

Verstraete, Garcia-Ripoll, Cirac, PRL 93, 207204 (2004)

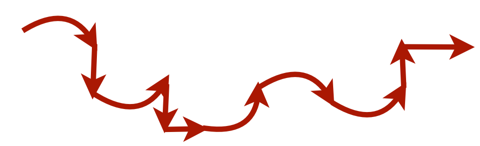
"Entanglement" entropy in MPDO

Operator Entanglement (**OE**)

Zanardi, Phys. Rev. A, 63, 040304 (2001)
Prosen, Pižorn, PRA 76, 032316 (2007)
J. Dubail, J. Phys. A: Math. Theor. 50 234001 (2017)
…

$$\hat{\rho}(t) \approx \sum_{\eta}^{N_t} p_\eta \, |\psi_\eta(t)\rangle \langle \psi_\eta(t)|$$

Stochastic evolution of MPS trajectories

$|\psi_\eta(0)\rangle$  $|\psi_\eta(t)\rangle$

Review: Daley, Adv. Phys. 63, 77 (2014)

$$|\psi_\eta\rangle \approx \mathrm{tr} \begin{pmatrix} |\psi_1^{11}\rangle & |\psi_1^{12}\rangle & \\ |\psi_1^{21}\rangle & |\psi_1^{22}\rangle & \\ & & \ddots \end{pmatrix} \begin{pmatrix} |\psi_2^{11}\rangle & |\psi_2^{12}\rangle & \\ |\psi_2^{21}\rangle & |\psi_2^{22}\rangle & \\ & & \ddots \end{pmatrix} \cdots \begin{pmatrix} |\psi_N^{11}\rangle & |\psi_N^{12}\rangle & \\ |\psi_N^{21}\rangle & |\psi_N^{22}\rangle & \\ & & \ddots \end{pmatrix}$$

Averaged "entanglement" entropy in MPS

Trajectory entanglement (**TE**)

Bonnes, Läuchli, arXiv:1411.4831
Nieuwenburg, Malo, Daley, Fischer, QST 01LT02 (2017)
Wolff, Sheikjan, Kollath, SciPost Phys. Core 3, 010 (2020)
Vovk, Pichler, PRL 128, 243601 (2022) …

Lecture 3 - Some recent results on entanglement dynamics in open spin-models

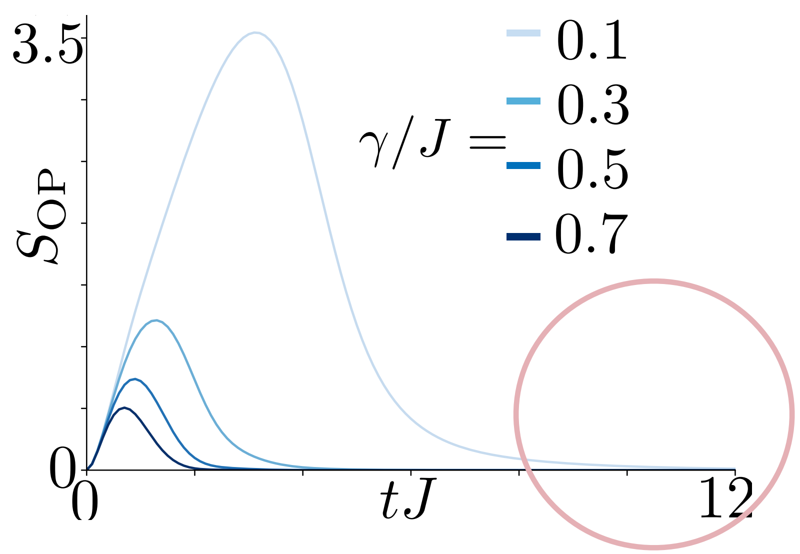
- **Model:** XXZ spin-model, highly excited initial Néel state

$$\frac{d}{dt}\hat{\rho} = -\mathrm{i}[\hat{H}_{XXZ}, \hat{\rho}] + \sum_i \mathcal{L}^{[i]}\hat{\rho}$$

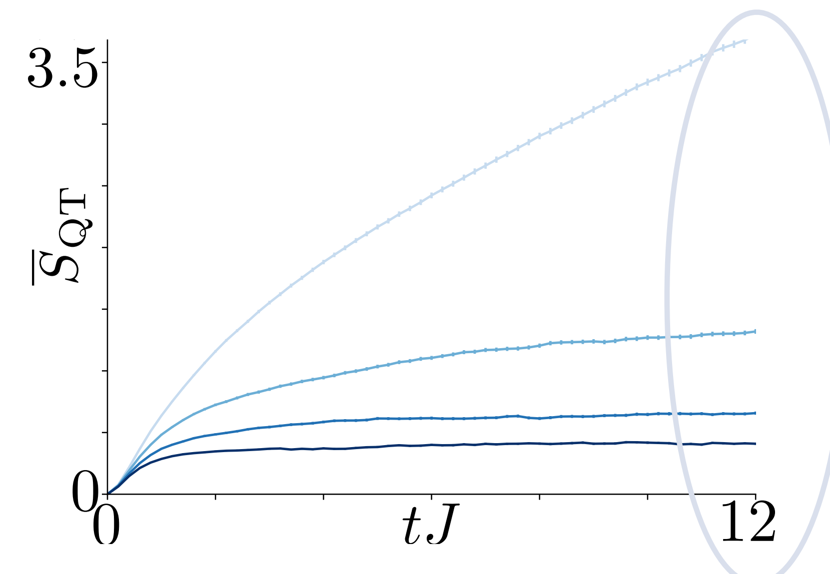
$$\mathcal{L}^{[i]}\hat{\rho} = \hat{L}_i \hat{\rho} \hat{L}_i^\dagger - \frac{1}{2}\{\hat{L}_i^\dagger \hat{L}_i, \hat{\rho}\},$$

Operator Entanglement (**OE**)

Trajectory entanglement (**TE**)



$$\hat{L}_i^+ = \sqrt{\gamma_+}\hat{\sigma}_i^+$$
$$\hat{L}_i^- = \sqrt{\gamma_-}\hat{\sigma}_i^-.$$
$$\gamma = \gamma^+ = \gamma^-$$



$t \to \infty \quad \hat{\rho} \to \prod_i \hat{\rho}^{[i]}$ $\quad \chi = 1$

$$\hat{\rho} \approx \sum_{\eta}^{N_t} p_\eta \left|\psi_\eta(t)\right\rangle \left\langle\psi_\eta(t)\right| \quad \chi \gg 1$$

fundamentally inefficient decomposition

- **For dephasing:** OE scales logarithmic, TE scales clearly super-logarithmic!

Ruben Daraban poster

G. Preisser, D. Wellnitz, T. Botzung, and JS, Phys. Rev. A 108, 012616 (2023)

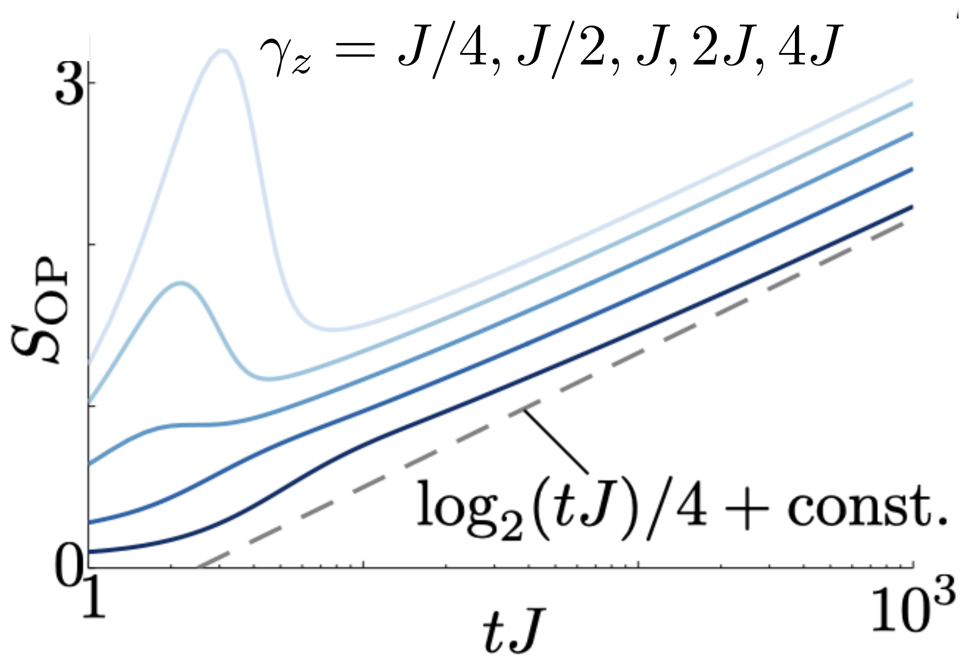
OE & dephasing: rise and fall, and slow rise again

- **Model:** XXZ spin-model, highly excited initial Néel state

$$\frac{d}{dt}\hat{\rho} = -\mathrm{i}[\hat{H}_{XXZ}, \hat{\rho}] + \sum_i \mathcal{L}^{[i]}\hat{\rho}$$

$$\mathcal{L}^{[i]}\hat{\rho} = \hat{L}_i\hat{\rho}\hat{L}_i^\dagger - \frac{1}{2}\{\hat{L}_i^\dagger\hat{L}_i, \hat{\rho}\},$$

$$\hat{H}_{\mathrm{XXZ}} = \frac{1}{4}\sum_i \left(J\hat{\sigma}_i^x\hat{\sigma}_{i+1}^x + J\hat{\sigma}_i^y\hat{\sigma}_{i+1}^y - J\hat{\sigma}_i^z\hat{\sigma}_{i+1}^z\right)$$



Noise: Dephasing $\hat{L}_i^z = \sqrt{\gamma_z}\hat{\sigma}_i^z$

Universal logarithmic long-time growths!

$$S_{\mathrm{OP}}(t) = S_0 + \eta\log(tJ) \qquad \eta \to \frac{1}{4}$$

- **Explanation:** *Consequence of magnetization conservation*

In large dissipation limit:
*Symmetry blocks in reduced density matrices follow **classical** (sub-)diffusion processes.*

- **In terms of numerical complexity:** $S_{\mathrm{OP}} \propto \log(t)$ $\max[S_{\mathrm{op}}] \propto \log(\chi)$ → $\chi \propto t$

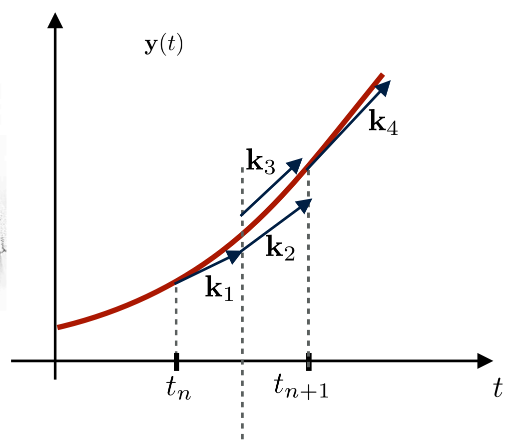
*Not necessarily hard to simulate **classically**.*

D. Wellnitz, G. Preisser, V. Alba, J. Dubail, and JS, Phys. Rev. Lett. 129, 170401 (2022)

Full recap

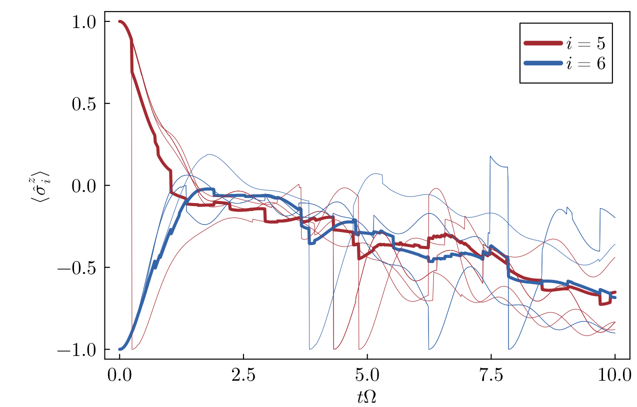
A tour through some numerical methods for simulating large quantum many-body non-equilibrium dynamics, with examples. Learning physics by simulating it.

- We discussed numbers and linear algebra on classical computers.



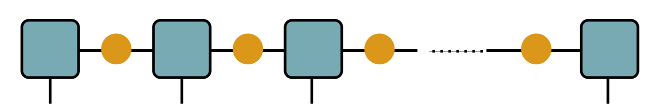
- We discussed several numerical methods for the simulation of quantum dynamics (both in close and open systems): **Runge-Kutta, Krylov space**

- A **mean-field approximation** (e.g. GP) can drastically **reduce the state-space** but makes the problem **non-linear** (and it's a strong approximation). Mean-field relies on non-entangled product states. It's an the limit of a matrix product state with bond dimension zero.



- For open systems we discussed **full density matrix representations** vs. **quantum trajectory approaches.** We introduced ways to construct **sparse** Hamiltonian and Liouvillian matrices (with kron)

- In the first two lectures, we kept the quantum state-representation **exact!** Then going to systems with more than 30 qubits/spins is hard.



- However, often information of the full state vector is overkill: Today we introduced **matrix product states**, a way to compress information! The crucial quantity that makes classical simulations hard is **not system size,** but entanglement entropy!

- Especially in open system: entanglement growth is therefore an interesting topic to look at, and far from understood ...